CHAPTER 2
PARAMETER OPTIMIZATION OF FIR FILTERS

2.1 Introduction

Among the different methods available for the design of non-recursive filters stated in chapter 1, windowing method is easy to apply and requires a relatively less amount of computation. However, it leads to sub-optimal designs whereby the filter order required to satisfy a set of given specifications is not the lowest that can be achieved. Consequently, the number of arithmetic operations required per output sample is not minimum and the computational efficiency and speed of operation of the filter are not as high as they could be [27].

In the weighted Chebyshev design, an error function is formulated for the desired filter in terms of a linear combination of cosine functions, and is then minimized by using a very efficient multivariable optimization algorithm known as Remez exchange algorithm. When convergence is achieved, the error function becomes equiripple. The amplitude of the error in different frequency bands is controlled by applying weights to the error function [27].

---

A part of the work reported in this chapter has the following publications:


This design method developed by Parks and McClellan (PM) using Remez exchange algorithm is simple, robust, flexible, fast, and computationally efficient and hence is the best Chebyshev approximation for designing a large class of optimum FIR linear phase digital filters. But the Remez exchange algorithm has the following limitations [20]:

1. When the restrictions are imposed simultaneously on both time and frequency responses of the filter, Remez cannot be applied.
2. In the design of interpolation filters, where some of the coefficients are constrained to be zero, the alternation theorem no longer applies and hence Remez cannot be applied.

In the above said situations, linear programming can be used but it also suffers from the limitation that it is much slower than Remez and hence restricted to filters of limited length only. Moreover, application of linear programming for the filter design requires some approximation to make the inherent, non-linear problem into a linear one.

Genetic Algorithm (GA) is a robust and efficient optimization procedure [74], capable of finding the global optimal solution to the filter design problems. In this chapter, a new algorithm using GA is proposed to optimize the FIR filter designed by windowing method. The parameters of the filter used for optimization are filter length and cut-off frequency. The performance of the proposed algorithm is compared with that of Parks and McClellan (PM) algorithm. First, the linear phase property of the filter is discussed and Weighted Chebyshev approximation problem is formulated. The solution to the approximation problem given by Parks and McClellan using Remez algorithm is discussed next. The proposed algorithm using GA for windowing method is dealt with. A comparison of GA based windowing with Remez algorithm has been made for various filter specifications.
2.2 Linear Phase Response and Its Implications

The ability to have an exact linear phase response is one of the most important properties of FIR filters. The necessary and sufficient condition for a filter to have a linear phase response is that its impulse response must be real, finite length, and symmetric about its mid point. Additionally, the filter should have a constant phase delay [75]. The positive and negative symmetry are stated in the Eqns. 2.1a and 2.1b respectively.

\[ h[n] = h[N-1-n], \quad 0 < n < N-1 \] ... (2.1a)
\[ h[n] = -h[N-1-n], \quad 0 < n < N-1 \] ... (2.1b)

where \( N \) is the number of coefficients of the filter. The condition on the phase response is given in Eqns. 2.2a and 2.2b.

\[ \theta(\omega) = -\alpha \omega, \quad -\pi \leq \omega \leq \pi \] ... (2.2a)

or

\[ \theta(\omega) = \beta -\alpha \omega, \quad -\pi \leq \omega \leq \pi \] ... (2.2b)

where \( \alpha \) is a constant phase delay in samples and \( \beta = \pm \pi/2 \). There are four types of linear-phase FIR filters, [76] depending on whether \( N \) is even or odd and whether \( h[n] \) has positive or negative symmetry. Fig.2.1 illustrates the impulse responses of the four types of linear phase FIR filters.
Fig. 2.1 Impulse responses of Linear Phase FIR Filter

Table 2.1 shows the expressions for the frequency response of the four types of linear phase FIR filters.
Table 2.1 Frequency Response Expressions

<table>
<thead>
<tr>
<th>Type of linear Phase</th>
<th>Impulse response Symmetry</th>
<th>Number of Coefficients N</th>
<th>Frequency response $H(e^{j\omega})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Positive symmetry</td>
<td>odd</td>
<td>$e^{-j\omega(N-1)/2} \sum_{n=0}^{(N-1)/2} a[n] \cos(\omega n)$</td>
</tr>
<tr>
<td>2</td>
<td>even</td>
<td>even</td>
<td>$e^{-j\omega(N-1)/2} \sum_{n=1}^{N/2} b[n] \cos(\omega (n-1/2))$</td>
</tr>
<tr>
<td>3</td>
<td>Negative symmetry</td>
<td>odd</td>
<td>$e^{-j[(\omega(N-1)/2)-\pi/2]} \sum_{n=1}^{(N-1)/2} c[n] \sin(\omega n)$</td>
</tr>
<tr>
<td>4</td>
<td>even</td>
<td>even</td>
<td>$e^{-j[(\omega(N-1)/2)-\pi/2]} \sum_{n=1}^{N/2} d[n] \sin(\omega (n-1/2))$</td>
</tr>
</tbody>
</table>

where $a[0] = h \frac{N-1}{2}$, $a[n] = 2h \frac{N-1}{2} - n = c[n]$, $b[n] = 2h \frac{N}{2} - n = d[n]$

The frequency response of type 2 filter is always zero at $\omega=\pi$ and hence is not suitable for a high pass filter. Type 3 and 4 filters introduce a phase shift of 90° and the frequency response is always zero at $\omega=0$ and so not suitable for low pass filters, but can be used to design differentiators and Hilbert transformers. Type 3 filters are also not suitable for high pass as their frequency response is zero at $\omega = \pi$. Type 1 is the most versatile filter used for low pass design. The phase delay for type 1 and 2 filters or group delay for all four types of filters can be expressed in terms of the number of coefficients of the filter and hence be corrected to yield a zero phase or group delay response. The phase delay $T_p$ for type 1 and 2 filters is given by

$$T_p = [(N-1) / 2] T \quad \text{... (2.3a)}$$

and for type 3 and 4 filters, the group delay $T_g$ is given by

$$T_g = [(N-1-\pi) / 2] T \quad \text{... (2.3b)}$$

where $T$ is the sampling interval.
2.2.1 FIR Filter Specifications

The magnitude response of an FIR low pass filter is specified in the form of a tolerance scheme shown in Fig. 2.2.

![Magnitude Response Specifications](image)

**Fig. 2.2 Magnitude Response Specifications**

The parameters are

- \( \delta_p \) = peak pass band deviation or ripple
- \( \delta_s \) = peak stop band ripple
- \( \omega_p \) = pass band edge frequency
- \( \omega_s \) = stop band edge frequency
- \( \Delta \omega \) = transition width

Some times, the maximum number of coefficients \( N \) that can be accepted may also be specified.

2.3 Chebyshev Approximation

An approximation that minimizes the maximum error over a set of frequencies is called a Chebyshev approximation [27], where the error is defined as the difference between the desired frequency response and the actual frequency response. Filters that have the minimum value of the maximum error
exhibit equiripple behaviour in their frequency responses. Thus, the optimum Chebyshev filters are called equiripple filters. The frequency response of the four types of linear phase filters [76] could be written in the form given in Eqn. (2.4)

\[ H(e^{j\omega}) = e^{-j\omega(N-1)/2}e^{j(\pi/2)L}H^*(e^{j\omega}) \]  

...(2.4)

where, the function \( H^*(e^{j\omega}) \) is purely real. The values for the constant \( L \) and the form for \( H^*(e^{j\omega}) \) are given in Table 2.2 for each of the four types of linear phase.

<table>
<thead>
<tr>
<th>Type of Linear Phase</th>
<th>( L )</th>
<th>( H^*(e^{j\omega}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>( \sum_{n=0}^{(N-1)/2} a[n]\cos[\omega n] )</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>( \sum_{n=1}^{N/2} b[n]\cos[\omega(n-1/2)] )</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>( \sum_{n=0}^{(N-1)/2} c[n]\sin[\omega n] )</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>( \sum_{n=1}^{N/2} d[n]\sin[\omega(n-1/2)] )</td>
</tr>
</tbody>
</table>

The expressions for \( H^*(e^{j\omega}) \) can be written as a product of \( Q(e^{j\omega}) \), a fixed function of \( \omega \) and a term \( P(e^{j\omega}) \), a sum of cosines. The values of these functions for all four types of linear phase filter are shown in Table 2.3.

<table>
<thead>
<tr>
<th>Type of Linear Phase</th>
<th>( Q(e^{j\omega}) )</th>
<th>( P(e^{j\omega}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>( \sum_{n=0}^{(N-1)/2} \tilde{a}[n]\cos[\omega n] )</td>
</tr>
<tr>
<td>2</td>
<td>( \cos[\omega/2] )</td>
<td>( \sum_{n=0}^{(N/2)-1} \tilde{b}[n]\cos[\omega n] )</td>
</tr>
<tr>
<td>3</td>
<td>( \sin[\omega] )</td>
<td>( \sum_{n=0}^{(N-3)/2} \tilde{c}[n]\cos[\omega n] )</td>
</tr>
<tr>
<td>4</td>
<td>( \sin[\omega/2] )</td>
<td>( \sum_{n=1}^{(N/2)-1} \tilde{d}[n]\cos[\omega n] )</td>
</tr>
</tbody>
</table>
To show how the optimal linear phase FIR filter design problem can be formulated as Chebyshev approximation problem, it is necessary to define \( D(e^{j\omega}) \), the desired real frequency response of the filter, and \( W(e^{j\omega}) \), a weighting function on the approximation error that enables the user to choose the relative size of the error in different frequency bands. The weighted error of approximation \( E(e^{j\omega}) \) is as given in Eqn. (2.5)

\[
E(e^{j\omega}) = W(e^{j\omega})[D(e^{j\omega}) - P(e^{j\omega})Q(e^{j\omega})]
\]  

... (2.5)

Defining \( \tilde{W}(e^{j\omega}) \) and \( \tilde{D}(e^{j\omega}) \) as shown in Eqns. (2.6 a) and (2.6 b),

\[
\tilde{W}(e^{j\omega}) = W(e^{j\omega})Q(e^{j\omega})
\]  

... (2.6a)

\[
\tilde{D}(e^{j\omega}) = \frac{D(e^{j\omega})}{Q(e^{j\omega})}
\]  

... (2.6b)

the error function becomes,

\[
E(e^{j\omega}) = \tilde{W}(e^{j\omega})[\tilde{D}(e^{j\omega}) - P(e^{j\omega})]
\]  

... (2.7)

The Chebyshev approximation problem is now stated as finding the set of coefficients \( \tilde{a}[n] \) and \( \tilde{b}[n] \), to minimize the maximum absolute value of \( E(e^{j\omega}) \) over the frequency bands in which the approximation is being performed. The optimization problem is mathematically stated as given in Eqn. (2.8)

\[
\|E(e^{j\omega})\| = \min_{a,b} \left[ \max_{\omega \in \mathcal{A}} |E(e^{j\omega})| \right]
\]  

... (2.8)

where, \( \mathcal{A} \) represents the disjoint union of all the frequency bands of interest. A solution to this Chebyshev approximation problem is given by alternation theorem [75], which is stated as follows:

**Alternation Theorem**: If \( P(e^{j\omega}) \) is a linear combination of \( r \) cosine functions, that is, \( P(e^{j\omega}) = \sum_{n=0}^{r-1} a[n] \cos(\omega n) \), then a necessary and a sufficient condition that \( P(e^{j\omega}) \) be the unique, best weighted Chebyshev approximation to a continuous function \( \tilde{D}(e^{j\omega}) \) on \( \mathcal{A} \), a compact subset of \((0, \pi)\), is that the weighted error function \( E(e^{j\omega}) \) exhibits at least \((r+1)\) extremal frequencies in \( \mathcal{A} \); that is
there must exist \((r+1)\) points \(\omega_i\) in \(A\) such that \(\omega_1, \omega_2, \ldots, \omega_{r+1}\) and such that \(E(e^{j\omega_i}) = -E(e^{j\omega_{i-1}}), i = 1, 2, \ldots, r\) and \(\max_{\omega \in A} \left| E(e^{j\omega}) \right| = \max_{\omega \in A} \left| E(e^{j\omega}) \right|.

### 2.3.1 Remez Exchange Algorithm

Remez exchange algorithm can be used to find the solution to the Chebyshev approximation problem [27],[75],[77]. The Remez exchange algorithm makes use of the fact that it is always possible to make the error function takes on the values \(\pm \delta\) for any given set of \(r+1\) frequency points. Fig.2.3 shows the block diagram representation of Remez algorithm. A dense grid of frequency points is used to find the set of \((r+1)\) extremal frequencies required by the alternation theorem. Given a set of these extremal frequencies \(\{\omega_k\}, k = 0, 1, \ldots, r\), the Remez exchange algorithm consists of the following four basic computations:

1. Solve the following set of equations to calculate \(\delta\):

   \[ \hat{W}(e^{j\omega_k})[\hat{D}(e^{j\omega_k}) - P(e^{j\omega_k})] = (-1)^k \delta, \quad k = 0, 1, \ldots, r \]

   \[ (2.9) \]

2. Use the Lagrange interpolation formula to interpolate \(P(e^{j\omega})\) on the \(r\) points \(\omega_0, \omega_1, \ldots, \omega_{r-1}\).

3. Evaluate the error function \(E(e^{j\omega})\) on the dense set of frequencies.

4. If \(\left| E(e^{j\omega}) \right| \leq \delta\) for all frequencies in the dense set, then the optimal approximation has been found. If \(\left| E(e^{j\omega}) \right| > \delta\) for some frequencies in the dense set, then a new set of \((r+1)\) frequencies must be chosen. The new points are chosen as the peaks of the resulting error curve, thereby forcing \(\delta\) to increase and ultimately converge to its upper bound, which corresponds to the solution to the problem.
If these points are more than \((r+1)\) extrema in \(E(e^{j\omega})\) at any iteration, the \((r+1)\) frequencies at which \(|E(e^{j\omega})|\) is largest are retained as the guessed set of extremal frequencies for the next iteration.

**Fig. 2.3 Remez Exchange Algorithm**
So, for a given set of filter specifications, a Chebyshev approximation problem using a weighted combination of cosines is formulated and the Remez exchange algorithm is used to find the extremal frequencies. When the extremal frequencies are found, the impulse response is determined from the frequency response. This design method developed by Parks and McClellan (PM) using Remez exchange algorithm is simple, robust, flexible, fast, and computationally efficient and hence is the best Chebyshev approximation for designing a large class of optimum FIR linear phase digital filters. But the Remez exchange algorithm has the following limitations [20]:

1. When the restrictions are imposed simultaneously on both time and frequency response of the filter, Remez cannot be applied.
2. In the design of interpolation filters, where some of the coefficients are constrained to be zero, the alternation theorem no longer applies and hence Remez cannot be applied.

In the above said situations, linear programming can be used but it also suffers from the limitation that it is much slower than Remez and hence restricted to filters of limited length only. Moreover, application of linear programming for the filter design requires some approximation to make the inherent, non-linear problem into a linear one.

Windowing method, which is a simple but sub-optimal method of designing FIR filters [78], is used in the proposed algorithm.

### 2.4 Windowing Method

If $k_p$, $\omega_p$ and $k_s$, $\omega_s$ represent the pass band and stop band attenuation frequency requirements of the digital low pass filter, an iterative procedure to design such a filter is shown in Fig.2.4.
Select the length of the window such that
\[ N \geq k \frac{2\pi}{\omega_s - \omega_p} \]
(k depends upon the type of window used)

Fix \( \omega_c = \omega_1 \) and \( \alpha = (N-1)/2 \) for the trial impulse response
\[ h[n] = \sin[\omega_c (n-(N-1)/2)] / \pi [n-(N-1)/2] \cdot w[n] \]

Reduce \( N \)

Check if a further reduction in \( N \) is possible

Yes

Plot the frequency response \( H(e^{j\omega}) \) \(^\circ\) and check if the given specifications are satisfied

No

Adjust \( \omega_c \), normally larger on the first iteration

\(^\circ\) - \[ H(e^{j\omega}) = e^{-j\omega (N-1)/2} \{ h((N-1)/2) + \sum_{n=0}^{(N-1)/2} 2h[n] \cos[\omega (n-(N-1)/2)] \} \]
for \( N \) odd

\[ H(e^{j\omega}) = e^{-j\omega (N-1)/2} \{ \sum_{n=0}^{N/2-1} 2h[n] \cos[\omega (n-(N-1)/2)] \} \]
for \( N \) even

Fig. 2.4 Windowing Method
This procedure is a trial and error method and performs satisfactorily. But this method is not an optimal one. Normally, a filter found using this procedure would have a higher N than what could be designed by other computer techniques [78].

Hence, in the windowing method, there is a need of optimizing the filter length N. It should be large enough, so that the error due to the approximation to ideal frequency response is minimized, and small enough, to allow reasonable implementation.

In the windowing method, the designed filter should first satisfy the magnitude response specifications. Once the satisfaction of the magnitude response is realized, and then it is necessary to check for the satisfaction of the magnitude response with a lesser N, which is a time domain parameter. In such situations, where, the restrictions are imposed simultaneously on both time and frequency responses of the filter, Remez cannot be applied [20]. In those cases, linear programming can be used, but it also suffers from the limitation that it is much slower than Remez and requires some approximation to make the inherent non-linear problem into a linear one [21],[22]. Since, the existing techniques for solving the filter design problems are having the above mentioned limitations, it has been decided to use Genetic Algorithm, which is a robust, and efficient optimization procedure capable of searching in parallel for the global optimum solution for the filter design problems [74].

2.5 GA Based Windowing Algorithm

As seen in the windowing method, the cut-off frequency is adjusted first for satisfying the magnitude response of the filter and then the filter length is reduced for the satisfaction of the magnitude response with lesser value of N. GA is applied for finding the optimal values of N and $\omega_c$. At each generation, the algorithm generates different values of N and $\omega_c$ for the given magnitude
frequency requirements of the filter. From these values, optimum value of N is found by fitness function evaluation.

By applying the optimization technique, the computational efficiency and speed of operation of the filtering system can be improved. The performance of the proposed algorithm for different filter specifications are studied and compared. It is necessary that the value of the filter length N be made optimum to the required specifications. The value of N shall be neither small nor large, as it would affect the efficiency of implementation of the filter in a Digital Signal Processor. Thus the following are the basic objectives of the optimization problem:

1. Reduction of Computational Complexity.
2. Minimizing the error between the desired and the practically obtained frequency response of the filter.
3. Decreasing the amount of memory utilization and processing time of the Digital Signal Processor.
4. Low implementation and design cost.

Many practical optimum design problems are characterized by mixed continuous-discrete variables, and discontinuous and non-convex design spaces. Genetic Algorithms (GAs) are well suited for solving such problems.

For minimization problems, its equivalent maximization problem is considered such that the optimum point remains unchanged. The fitness function, F(x) is expressed as,

\[ F(x) = \frac{1}{1 + f(x)} \]  

... (2.10)

where, f(x) is the objective function to be minimized. This transformation does not alter the location of the minimum, but converts a minimization problem to an equivalent maximization problem.
By measuring the error in frequency response as shown by the procedure below, the fitness function is decided and by recombination and selection process the search is established throughout the search space. Let $N^*, \omega_c^*$ be the current design vector variables, then the objective (error) function $e\left(N^*, \omega_c^* \right)$ for this design vector is found by the procedure given below:

**Step 1: Initialization**

Initialize error function: $e\left(N^*, \omega_c^* \right) = 0$.

Set filter length $N = N^*$ and Cut-off frequency $\omega_c = \omega_c^*$

**Step 2: Filter design.**

Find the impulse response $h(n)$ of the filter for $n = 0$ to $N-1$ using the equation

$$h[n] = \frac{\sin[\omega_c (n - \frac{N-1}{2})]}{\pi[n - \frac{N-1}{2}]} . w[n] \quad \cdots (2.11)$$

**Step 3: Error in Pass-band.**

Calculate $H(e^{j\omega})$, from $\omega = 0$ to $\omega = \omega_p$ using the equation

$$H\left(e^{j\omega} \right) = e^{-j\omega \frac{(N-1)}{2}} \left\{ h[\frac{(N-1)}{2}] + \sum_{n=0}^{(N-3)} 2h[n] \cos[\omega (n - \frac{N-1}{2})] \right\} \quad \cdots (2.12)$$

Find $\Delta k = 20 \log_{10}(|H(e^{j\omega})| - |k_1|) \quad \cdots (2.13)$

If $\Delta k > 0$, then accumulate $\Delta k$ to error function value. $e(N^*, \omega_c^*) = e(N^*, \omega_c^*) + \Delta k$.

Otherwise increment $\omega$ and go to step 3.

**Step 4: Error in Stop-band.**

Calculate $H(e^{j\omega})$, from $\omega = \omega_s$ to $\omega = \pi$ using (2.43)

Find $\Delta k = |k_2| - 20 \log_{10}(|H(e^{j\omega})|) \quad \cdots (2.14)$

If $\Delta k > 0$, then accumulate $\Delta k$ to error function value. $e(N^*, \omega_c^*) = e(N^*, \omega_c^*) + \Delta k$.

Otherwise increment $\omega$ and repeat step 4.
The objective function is the accumulated $e(N^*,\omega_c^*)$, the value of the error function, for the current design vector $(N',\omega_c')$, which is to be minimized.

2.6 Implementation of the Algorithm

The proposed algorithm is implemented as given below:

- Create population of 20 samples from search space.
- Each chromosome is made of two variables, the filter length, and the cut-off frequency.
- A binary string of length 40 is used to form the chromosome. The first 20 bits represent the variable filter length $N$ and the last 20 represent the cutoff frequency $\omega_c$.
- Calculate the cost function and assign the available fitness to each individual.
- Perform crossover by taking 2 chromosomes at an instant with the crossover probability 0.8.
- Mutate the newly generated offspring with the mutation probability 0.2.
- The generation gap is taken as 0.9. This is to have elitist strategy, which retains 10% of the best individuals of the previous generation.
- Repeat the above steps to obtain the optimized results.

2.7 Simulation Results and Discussion

The parameters taken into consideration for optimization are the filter length $N$ and the cut-off frequency $\omega_c$. The low pass filters with specifications as shown in Table 2.4 are considered for optimizing the above mentioned parameters using GA. The optimized values of $N$ and $\omega_c$ are shown in Table 2.5. From the filter length obtained by the proposed and Remez algorithm given in Table 2.6, it is observed that, the proposed algorithm using GA gives smaller filter lengths than Remez. The comparison is also represented in Fig. 2.5 in the form of a Bar chart.
<table>
<thead>
<tr>
<th>Filter</th>
<th>$\Omega_p$ (rad/s)</th>
<th>$K_1$ (dB)</th>
<th>$\Omega_s$ (rad/s)</th>
<th>$K_2$ (dB)</th>
<th>$\omega_p$ (rad/sample)</th>
<th>$\omega_s$ (rad/sample)</th>
<th>Sampling rate (samples/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1[81]</td>
<td>1000 $\pi$</td>
<td>- 0.1</td>
<td>1500 $\pi$</td>
<td>- 44</td>
<td>0.4 $\pi$</td>
<td>0.6 $\pi$</td>
<td>2500</td>
</tr>
<tr>
<td>2[81]</td>
<td>300 $\pi$</td>
<td>- 0.1</td>
<td>500 $\pi$</td>
<td>- 40</td>
<td>0.3 $\pi$</td>
<td>0.5 $\pi$</td>
<td>1000</td>
</tr>
<tr>
<td>3[82]</td>
<td>1000 $\pi$</td>
<td>- 1</td>
<td>1200 $\pi$</td>
<td>- 20</td>
<td>0.5 $\pi$</td>
<td>0.6 $\pi$</td>
<td>2000</td>
</tr>
<tr>
<td>4[76]</td>
<td>20 $\pi$</td>
<td>- 0.026</td>
<td>40 $\pi$</td>
<td>- 30</td>
<td>0.1563 $\pi$</td>
<td>0.3125 $\pi$</td>
<td>128</td>
</tr>
<tr>
<td>5[76]</td>
<td>20 $\pi$</td>
<td>- 0.026</td>
<td>40 $\pi$</td>
<td>- 30</td>
<td>0.0781 $\pi$</td>
<td>0.1563 $\pi$</td>
<td>256</td>
</tr>
<tr>
<td>6[76]</td>
<td>2000 $\pi$</td>
<td>- 0.026</td>
<td>3000 $\pi$</td>
<td>- 50</td>
<td>0.25 $\pi$</td>
<td>0.375 $\pi$</td>
<td>8000</td>
</tr>
<tr>
<td>7[76]</td>
<td>1000 $\pi$</td>
<td>- 0.01</td>
<td>2000 $\pi$</td>
<td>- 40</td>
<td>0.1 $\pi$</td>
<td>0.2 $\pi$</td>
<td>10000</td>
</tr>
<tr>
<td>8[83]</td>
<td>200 $\pi$</td>
<td>-0.1737</td>
<td>300 $\pi$</td>
<td>- 40</td>
<td>0.2 $\pi$</td>
<td>0.3 $\pi$</td>
<td>1000</td>
</tr>
<tr>
<td>9[84]</td>
<td>350 $\pi$</td>
<td>-0.1737</td>
<td>450 $\pi$</td>
<td>- 40</td>
<td>0.35 $\pi$</td>
<td>0.45 $\pi$</td>
<td>1000</td>
</tr>
<tr>
<td>10[78]</td>
<td>200 $\pi$</td>
<td>- 2</td>
<td>1000 $\pi$</td>
<td>- 20</td>
<td>0.05 $\pi$</td>
<td>0.25 $\pi$</td>
<td>4000</td>
</tr>
</tbody>
</table>
### Table 2.5 Optimization Results using GA

<table>
<thead>
<tr>
<th>Filter</th>
<th>N</th>
<th>$\omega_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>17</td>
<td>0.42</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>0.306</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
<td>0.515</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>0.191</td>
</tr>
<tr>
<td>5</td>
<td>38</td>
<td>0.092</td>
</tr>
<tr>
<td>6</td>
<td>23</td>
<td>0.266</td>
</tr>
<tr>
<td>7</td>
<td>28</td>
<td>0.102</td>
</tr>
<tr>
<td>8</td>
<td>32</td>
<td>0.2065</td>
</tr>
<tr>
<td>9</td>
<td>31</td>
<td>0.3544</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>0.138</td>
</tr>
</tbody>
</table>

### Table 2.6 Comparison of N by GA and Remez

<table>
<thead>
<tr>
<th>FILTER</th>
<th>$N_{\text{Remez}}$</th>
<th>$N_{\text{GA}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>23</td>
<td>17</td>
</tr>
<tr>
<td>2</td>
<td>21</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>18</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>29</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>57</td>
<td>38</td>
</tr>
<tr>
<td>6</td>
<td>46</td>
<td>23</td>
</tr>
<tr>
<td>7</td>
<td>56</td>
<td>28</td>
</tr>
<tr>
<td>8</td>
<td>40</td>
<td>32</td>
</tr>
<tr>
<td>9</td>
<td>40</td>
<td>31</td>
</tr>
<tr>
<td>10</td>
<td>7</td>
<td>5</td>
</tr>
</tbody>
</table>
Fig. 2.5 Bar chart for N by GA and Remez

The magnitude responses obtained for the filters specified in Table 2.4, by the proposed and Remez exchange algorithm are shown in Fig. 2.6. The plots on the right correspond to Remez and that on the left correspond to GA. It is observed from Fig. 2.6, that there is a widening of the pass band by Remez. In addition, it is also noted that when the pass band tolerance of the filters is restricted to 0.1 to 0.2 db, higher stop band attenuation is realized using GA.
Fig. 2.6 Magnitude Responses by GA and Remez
Fig. 2.6 Magnitude Responses by GA and Remez (Contd...)
Fig. 2.6 Magnitude Responses by GA and Remez (Contd...)

7. N= 28
8. N= 32
9. N= 31

GA

Remez

N= 56
N= 40
N= 40
The performances of the two algorithms in pass band, stop band, and transition band have been compared in Table 2.7. The values given in the table are the pass band and transition width with $\pi$ normalized to 1 and the stop band attenuation in dB.
TABLE 2.7 Performance Comparisons of Filters by GA and Remez

<table>
<thead>
<tr>
<th>Filter</th>
<th>Pass band width</th>
<th>Stop band attenuation</th>
<th>Transition width</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GA</td>
<td>Remez</td>
<td>GA</td>
</tr>
<tr>
<td>1</td>
<td>0.25</td>
<td>0.4</td>
<td>51.45</td>
</tr>
<tr>
<td>2</td>
<td>0.11</td>
<td>0.295</td>
<td>42.9</td>
</tr>
<tr>
<td>3</td>
<td>0.425</td>
<td>0.498</td>
<td>26.86</td>
</tr>
<tr>
<td>4</td>
<td>0.018</td>
<td>0.157</td>
<td>28.93</td>
</tr>
<tr>
<td>5</td>
<td>0.007</td>
<td>0.079</td>
<td>27.74</td>
</tr>
<tr>
<td>6</td>
<td>0.105</td>
<td>0.206</td>
<td>27.84</td>
</tr>
<tr>
<td>7</td>
<td>0.004</td>
<td>0.101</td>
<td>34.2</td>
</tr>
<tr>
<td>8</td>
<td>0.117</td>
<td>0.2</td>
<td>42.3</td>
</tr>
<tr>
<td>9</td>
<td>0.262</td>
<td>0.35</td>
<td>41.26</td>
</tr>
<tr>
<td>10</td>
<td>0.15</td>
<td>0.101</td>
<td>5.95</td>
</tr>
</tbody>
</table>

Based on the analysis of the results provided in Table 2.7, the following observations have been made:

- For the given filter specifications, the pass band width is more in Remez than that in GA, which implies that Remez satisfies pass band specifications very closely. But, this satisfaction is met with a higher value of N than that required by GA. On an average, pass band width of the filter designed by Remez is about 0.0939 more than that by GA at a cost of 14 more coefficients.

- Looking at the stop band, the attenuation level attained by GA is more than that by Remez for the filters with the pass band
tolerance in the range 0.1 to 0.2 dB. When the tolerance level is less than 0.1 or above 0.2, the stop band attenuation obtained using GA is comparable with that of Remez.

- The enhancement of the pass band by Remez results in less transition width.
- The performance of the filter by GA is better than Remez in terms of stop band attenuation and the number of filter coefficients. As GA results in a commendable reduction of filter coefficients, the implementation of the filter in real time, needs smaller number of delays, multipliers and adders than that would be required by Remez.
- Even though Remez covers a wide range of filters, when the pass band tolerance is in the range 0.1 to 0.2 dB, GA performs better than Remez, by attaining higher stop band attenuation with smaller length.

2.7.1 Application with Audio Codec

Another illustration of the proposed algorithm is for the design of a filter to meet the specifications of Audio Codec given in [30]. The stop band (SB) attenuation for four different levels specified as Level 0, Level 1, Level 2, and Level 3 are considered. The pass-band ripple for all four levels is taken as 0.25 db. The approximate stop band attenuations in decibels are 74 for level 0, (a noisy sound signal period), 36.4 for level 1, 24.2 for level 2 and 17.3 for level 3 (a clear sound signal). The filter length N and SB attenuation obtained by the proposed algorithm is compared with those by CSD representation and Remez algorithm. The values are given in Table 2.8.
Table 2.8 Comparison of SB attenuation

<table>
<thead>
<tr>
<th>CODEC FILTER</th>
<th>Required Stop Band (SB) attenuation (db)</th>
<th>CSD</th>
<th>Proposed Algorithm</th>
<th>REMEZ</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
<td>Found SB (db)</td>
<td>N</td>
<td>Found SB (db)</td>
</tr>
<tr>
<td>LEVEL 0</td>
<td>74</td>
<td>31</td>
<td>74</td>
<td>67</td>
</tr>
<tr>
<td>LEVEL 1</td>
<td>36.4</td>
<td>29</td>
<td>40.8</td>
<td>36</td>
</tr>
<tr>
<td>LEVEL 2</td>
<td>24.2</td>
<td>27</td>
<td>28.4</td>
<td>29</td>
</tr>
<tr>
<td>LEVEL 3</td>
<td>17.3</td>
<td>17</td>
<td>20.5</td>
<td>12</td>
</tr>
</tbody>
</table>

From Table 2.8, it is inferred that for Level 0, CSD meets the specified SB attenuation with N=31, whereas the proposed algorithm gives an attenuation of 74.45 db with N=67 and Remez gives an attenuation of 79.11 db with N=73. This Level 0 is considered to be noisy sound period. At Level 1, the proposed method gives an improvement of 1.66 db over the required attenuation level with N=36, whereas CSD method requires only 29 coefficients for 40.8 db attenuation and Remez needs 45 coefficients for attaining 38.85 db attenuation. At Level 2, GA results in an attenuation of 25.9 with N=29, which is less than the length of Remez by 7, and more than that of CSD by 2. For Level 3, GA requires only 12 coefficients, but gives an attenuation of only 12.67db, which is 4.63 db less than the required attenuation. For the three levels from Level 0 to 2, the proposed method satisfies the required stop band attenuation with a filter length, less than that of Remez and more than that of CSD. At Level 3, GA results in a filter of least length. But the stop band attenuation level is not met with. As the pass band tolerance of the filter is not given, a tolerance of 0.25 db is assumed. As GA satisfies the first three levels, the assumed tolerance is correct. The reason for GA not satisfying the Level 3 specification might be due to a higher tolerance in the pass band, whereas it is assumed to be a small value.

46
Among the three methods, even though the CSD filter requires least length, the pass band width is not specified by this method. The widening of pass band results only in Remez filter. Hence it is observed that, the proposed algorithm satisfies the stop band specification of the filter with a less number of coefficients than Remez. The number of coefficients obtained by GA is more than that of CSD representation. As the CSD method has not specified the pass band tolerance, its performance in terms of less \( N \) and good satisfaction of the stop band cannot be appreciated. So, it is concluded, that GA performs better than Remez, by satisfying the stop band attenuation with a less value of \( N \).

The magnitude response of the Codec filter using CSD representation [30] is shown in Fig.2.7. The magnitude responses of the same filter obtained using Remez Exchange algorithm for level 0, level 1, level 2, and level 3 are shown in Fig. 2.8(a), Fig. 2.9(a), Fig. 2.10(a), and Fig. 2.11(a) respectively. The magnitude responses of Codec filter for level 0, level 1, level 2, and level 3 obtained using GA are shown in Fig. 2.8 (b), Fig. 2.9(b), Fig. 2.10(b), and Fig. 2.11(b) respectively.
Fig. 2.8 (a) Magnitude Response of Level 0 using Remez Algorithm

Fig. 2.8 (b) Magnitude Response of Level 0 using GA

Fig. 2.9 (a) Magnitude Response of Level 1 using Remez Algorithm

Fig. 2.9 (b) Magnitude Response of Level 1 using GA
Fig. 2.10(a) Magnitude Response of Level 2 using Remez Algorithm

Fig. 2.10 (b) Magnitude Response of Level 2 using GA

Fig. 2.11 (a) Magnitude Response of Level 3 using Remez Algorithm

Fig. 2.11 (b) Magnitude Response of Level 3 using GA
It is observed from the plots of magnitude responses of Codec filter obtained using the two methods, that the pass band width of the filters obtained using Remez is more than that obtained using GA. The stop band attenuation of the filters by GA is comparable with that of Remez; GA realizes this attenuation with a lesser value of N.

2.8 Summary

In this chapter, a GA based windowing algorithm has been proposed for improving the sub optimality of the windowing method. The filter parameters considered for optimization are filter length and cut-off frequency. The performance of the proposed algorithm has been compared with that of the Parks and McClellan algorithm, which uses Remez exchange algorithm and found to yield filters of lesser length. The pass band performance of the filters is better in Remez. But, the attenuation in stop band is more in filters designed using GA, when the pass band tolerance is restricted to a range of 0.1 to 0.2 db. For the filters with the pass band tolerance less than 0.1 or more than 0.2 db, Remez gives higher stop band attenuation.

For the Codec filter also, the pass band is enhanced by Remez algorithm. The GA filter has comparable stop band attenuation and also lesser length than that by Remez filter. Hence, it is concluded that the proposed algorithm results in a filter of lesser length and higher stop band attenuation, for the filters whose pass band tolerance is restricted to 0.1 to 0.2 db.

In this chapter, GA is used to improve the performance of the windowing method, which is the time domain design of FIR filters, and the optimum values of filter length and cut-off frequency are found. The frequency sampling technique designs the filter by taking N samples of the frequency response, among which, the samples in the transition band are not specified. These coefficients are to be optimized for higher stop band attenuation. The next chapter deals with the optimization of frequency sampling design.