telecommunications &
signal processing
laboratory

# FIR Filters:

# Frequency-Weighted and Minimum-Phase Designs

***Peter Kabal***

Department of Electrical & Computer Engineering

McGill University

McGill

Version 1b: 2004-09-21

Version 1.5: 2007-11-22

# Table of Contents

# FIR Filters:
# Frequency-Weighted and Minimum-Phase Designs

This report discusses the use of weighting functions in the design of linear-phase FIR filters. It also considers the design of minimum-phase filters using constrained weighted equiripple approximations. It is shown that minimum-phase filters are useful not only because they have a smaller group delay than linear-phase filters, but also because for a given number of filter coefficients, they produce a better amplitude approximation.

## 1 Equip-ripple Linear-Phase FIR Filter Design

Equiripple linear-phase FIR filter design has become a mainstay of FIR filter design after the classic work by McClellan and Parks [1]. The linear-phase property ensures that the frequency response of the filter can be written as a phase factor (linear-phase) in cascade with a real frequency response which can be expressed as the sum of cosines. The sum of cosines term in turn can be expanded as a sum of cosine powers, i.e. a Chebyshev polynomial in $\cos(\omega)$ . With this decomposition, algorithms such as the Remez exchange procedure can be used to design optimal min-max approximations to a desired response.

Due to the availability of efficient design algorithms, linear-phase filters seem to be the filter type of choice for FIR filters. In this report, we discuss frequency-weighting for such designs. We also discuss the design of minimum-phase filters using a procedure which uses a linear-phase design as its kernel. Minimum-phase FIR filters, and more generally non-linear phase FIR filters, deserve more attention. They reduce the input/output delay due to the minimum-phase property. But, even more importantly, they provide a better approximation to a desired amplitude response than linear-phase filters; or provide the same goodness of approximation with a less complex filter. There seems to be a fear of non-linear phase. But there are many applications where the non-linear phase is not a handicap since the group delays variations in the passband are relatively small. It is always to be remembered that analog filter design always involves non-linear phase.

### 1.1 Linear-Phase FIR Filters

Consider a symmetric filter with an $N$ -term ( $N$ odd) impulse response,

$$h[n]=h[N-1-n], \qquad \text{for } 0 \le n \le N-1. \tag{1}$$

The frequency response for this filter can be written as (see Appendix A),

$$H(\omega) = e^{-j\omega(N-1)/2}\tilde{Q}(\omega)B^0(\omega), \tag{2}$$

where $B^0(\omega)$ is a real-valued response, the so-called zero-phase factor, and $\tilde{Q}(\omega)$ is a fixed response. For the purposes of designing the filter, we ignore the linear phase term and concentrate on designing the filter $B^0(\omega)$. Also $\tilde{Q}(\omega)$ is either purely real or purely imaginary. In the latter case, we can define a phase factor $e^{j\pi/2}$ to be lumped with the linear-phase term (giving a so-called generalized linear phase). Then the overall zero-phase response can be written as

$$H^0(\omega) = Q^0(\omega)B^0(\omega). \tag{3}$$

It is this zero-phase response, $B^0(\omega)$, which will be designed using the McClellan-Parks algorithm.

## 1.2 Filter Design Programs: DFiltFIR / DFiltMPFIR

The filter design routines described here have their genesis in the Fortran code published with the original McClellan-Parks algorithm [1]. This code was subsequently modularized to be able to be called from other routines. A flexible spectrum / weight modelling using monotonic cubic interpolants was introduced [2]. These interpolants have the desirable property that they are monotonic between the specified points. Thus, for instance, a positive weight is guaranteed to remain positive in between specified values. As many points as desired can be specified to approximate arbitrary functions.

This flexible interpolation allowed for the iterative design of Nyquist filters [3]. In that procedure, the weights for the stopband part of the filter are derived from the actual frequency response of the passband part of the filter. Finally, the constrained optimization procedure described by Grenez [4] was implemented. It is these constraints that will allow for the design procedure for linear-phase equiripple FIR filters to be used to generate minimum-phase FIR filters.

Later the design routines were ported to the C-language, allowing for cleaner interfaces using structures. The final step was the porting of the code to Matlab. This became feasible when computing speeds allowed for the use of an interpreted language such as Matlab. The ease of vector / matrix programming, together with a good support library makes this the language of choice for filter design programs.

The program DFiltFIR designs frequency-weighted linear-phase FIR filters. The program DFiltMPFIR designs frequency-weighted minimum-phase FIR filters.[1]

## 1.3 Equiripple Design Paradigm

Given a real desired response $D(\omega)$ defined on a subset of frequencies $\Omega_D$ and a positive error weighting function $W(\omega)$ defined on the same set of frequencies, the goal is to design a filter with controlled deviations from the desired response. The deviation is

$$\begin{aligned} E(\omega) &= W(\omega)[D(\omega) - H^0(\omega)] \\ &= W(\omega)Q^0(\omega)[\frac{D(\omega)}{Q^0(\omega)} - B^0(\omega)] \\ &= W_M(\omega)[D_M(\omega) - B^0(\omega)]. \end{aligned} \tag{4}$$

This equation shows that the term $Q^0(\omega)$ can be absorbed into the error weighting function and the desired response.

As shown in Appendix A, the zero-phase function $B^0(\omega)$ can be expressed as the sum of cosines with harmonically related frequencies, $\cos(\omega n)$. Further this sum can be expressed as a polynomial in $x = \cos(\omega)$. This is a Chebyshev polynomial expansion. During the design process, the polynomial is represented by its sample values, using a Lagrange interpolation procedure to evaluate the polynomial at other frequencies. At the end of the design process, by using the Remez exchange algorithm, the sample values of the polynomial will coincide with extrema of the error. The availability of the locations of these extrema will help us in extracting the minimum-phase component of the linear-phase design.

## 1.4 Minimum-Phase Filters

Decomposition of a linear-phase filter into a minimum-phase filter and a maximum-phase filter, where each has the same magnitude response, requires that the unit-circle zeros be double-order. A more detailed discussion appears in Appendix B. In the procedure used here, the double-zeros are ensured by applying constraints on the zero-phase response. Normally in the stopband, the zero-phase response oscillates around zero. By not allowing the zero-phase response to go

[1] The filter design programs are available from WWW-MMSP.ECE.McGill.CA/Documents.

negative in the stopband, the oscillations in the stopband will just touch zero. The touching points are places where there are double-zeros in the response.

If the stopband response is equiripple, an alternative is to use a lifting procedure which shifts the response upwards. This is described briefly in Appendix C. The constrained stopband approach is, however, much more flexible in not requiring an equiripple stopband.

After designing a constrained linear-phase filter, that filter must be factored into a minimum-phase and maximum-phase part. This involves root-finding, and sorting of the roots as described in Appendix B. That Appendix gives an efficient and accurate process for constructing the filter from its roots.

# 2 Example FIR Filter Designs

Consider the following half-band highpass filter design. The filter is to be applied to speech sampled at 16 kHz and is to be part of a system to measure the energy in the highband spectrum (4–8 kHz). The stopband is from 0–3850 Hz and the passband is from 4150–8000 Hz. The passband ripple for each of the subsequent designs will be $\pm 0.2$ dB.

## 2.1 Linear-Phase FIR Filter Design

A conventional linear-phase FIR design with 101 taps was designed. The stopband and passband have constant weighting. The relative weighting between the stopband and passband was set to be 4.5 to achieve the passband deviation of $\pm 0.2$ dB.[2] The stopband attenuation for this design is about 46 dB, see Fig. 1.

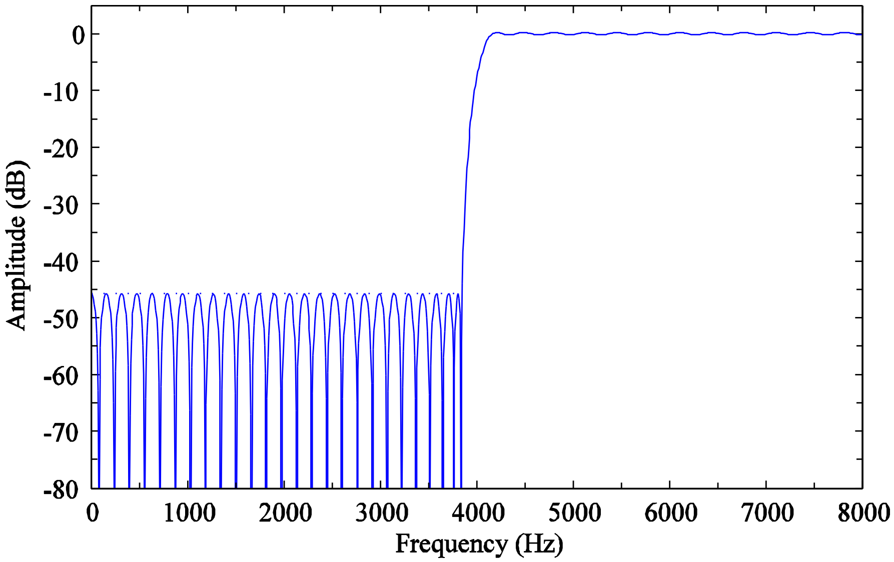

Fig. 1 Linear-phase FIR filter ( $N = 101$ ) with constant weighting in the stopband and passband.

Consider the long term average spectrum of speech [11] as shown as the top curve in Fig. 2. The middle curve in the figure shows the speech spectrum after filtering by the highpass filter. One notices that because of the large dynamic range between the low and high frequencies in a speech signal, the output of the filter will have significant leakage from the high amplitude low

[2] Appendix E gives the Matlab code used to design this filter and the subsequent examples.

frequencies. The margin between the lowest point in the stopband (at 8000 Hz) to the highest point in stopband (at low frequencies) is only about 14 dB.

## 2.2 Frequency-Weighted Linear-Phase FIR Filter Design

A modified filter design using a frequency-dependent weighting function in the stopband is shown in Fig. 3. The low frequencies are now more attenuated. For this design, the weighting varies linearly from the stopband edge to DC. The weight at DC is 10 times larger than the value at the stopband edge. This gives an extra 20 dB attenuation at DC relative to the stopband edge. The passband weight was chosen to give the same $\pm 0.2$ dB ripple as for the earlier filter. The extra attenuation at DC does not come for free — there is a little less attenuation near the cutoff frequency. The dotted line in the figure is at the attenuation level of the reference design with constant stopband weighting. We can see that there has been a tradeoff: a lot more attenuation at DC for a little less attenuation near the stopband edge. The price-to-pay is small. This is an example of the observation that it is far easier to get attenuation far from the transition region than it is at the passband edge.

The effect of the extra attenuation at low frequencies on the long-term spectrum of speech is shown as the lowest curve in Fig. 2. A measurement shows that the extra attenuation reduces the energy of the long-term spectrum of speech in the stopband by a factor of 27 relative to the filter with constant stopband attenuation.

## 2.3 Minimum-Phase FIR Filter Designs

A minimum-phase filter with the same number of taps as the previous two examples was designed. The weights were chosen to give the same 0.2 dB passband ripple. A frequency weighted stopband was used. The frequency response is shown in Fig. 4. The stopband deviation varies linearly for the minimum-phase filter. See Appendix D for a discussion of the weights of the minimum-phase filter needed to achieve this. This filter has a stopband attenuation which is 12 dB better than the comparable linear-phase filter. It can be noticed that the passband, though having the same ripple amplitude as the linear-phase filter, has twice as many ripples. The number of stopband ripples is the same. See Appendix C for a more extensive discussion.

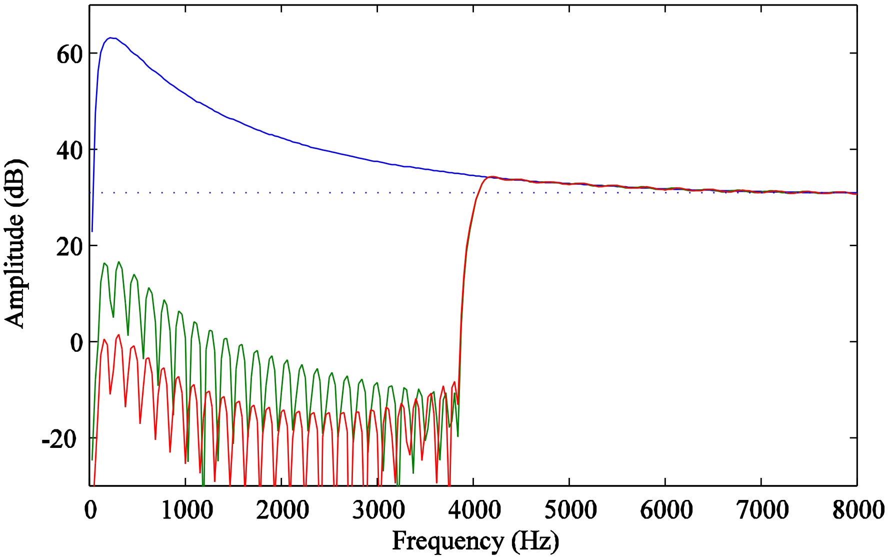

Fig. 2 Effects of filtering on the long-term spectrum of speech: Long-term spectrum of speech (top); long-term spectrum of speech filtered with a highpass filter with constant stopband weighting (middle); long-term spectrum of speech filtered with a highpass filter with a frequency-dependent weighting.

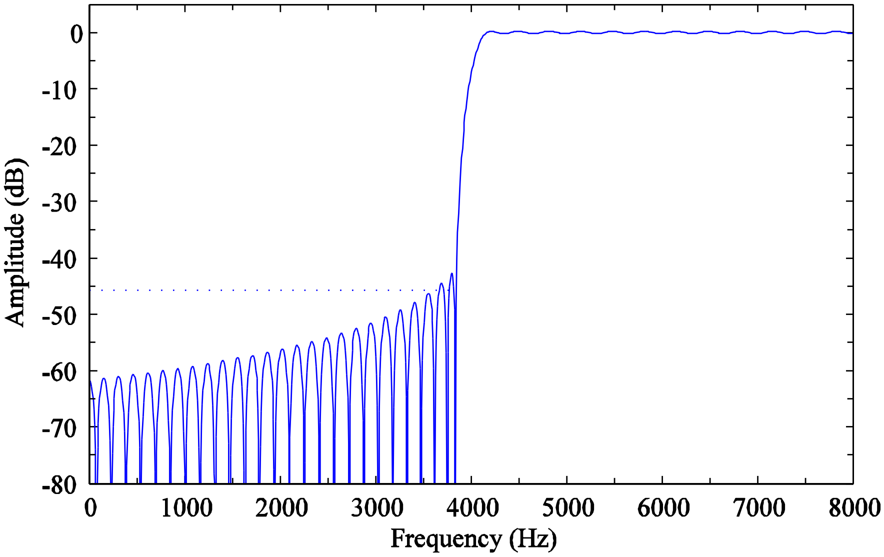

Fig. 3 Linear-phase filter ( $N = 101$ ) with frequency-dependent stopband weighting.

A second minimum-phase filter was also designed. The number of coefficients and the weights were chosen to give a passband ripple of $\pm 0.2$ dB and a stopband response comparable to the second linear-phase filter. The number of coefficients is 85, representing a saving of about 15% relative to the filters with 101 coefficients. The frequency response is shown in Fig. 5.

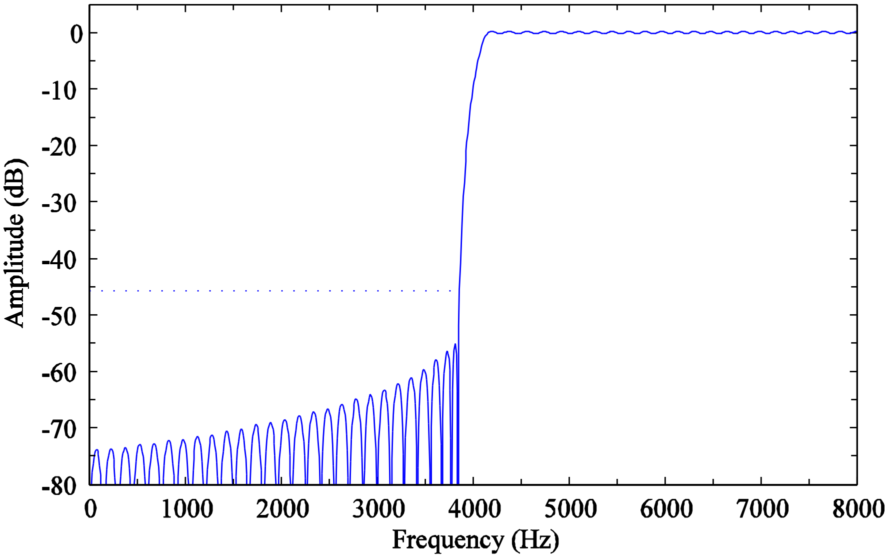

Fig. 4 Minimum-phase filter with 101 coefficients.

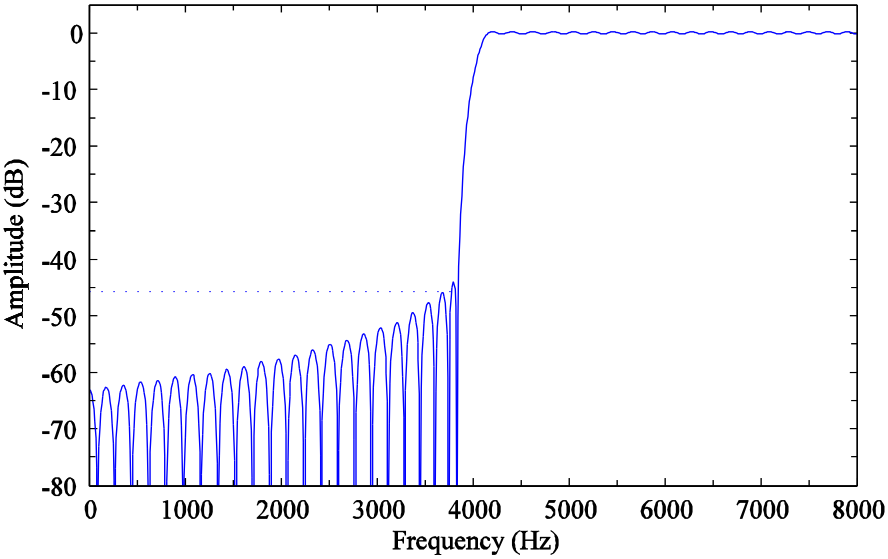

Fig. 5 Minimum-phase filter with 85 coefficients.

The group delay for the linear-phase filters is constant at $(N-1)/2$ samples. For the filters under consideration, this is 50 samples. The group delay for the minimum-phase filter is much smaller in the passband, but not constant, see Fig. 6.

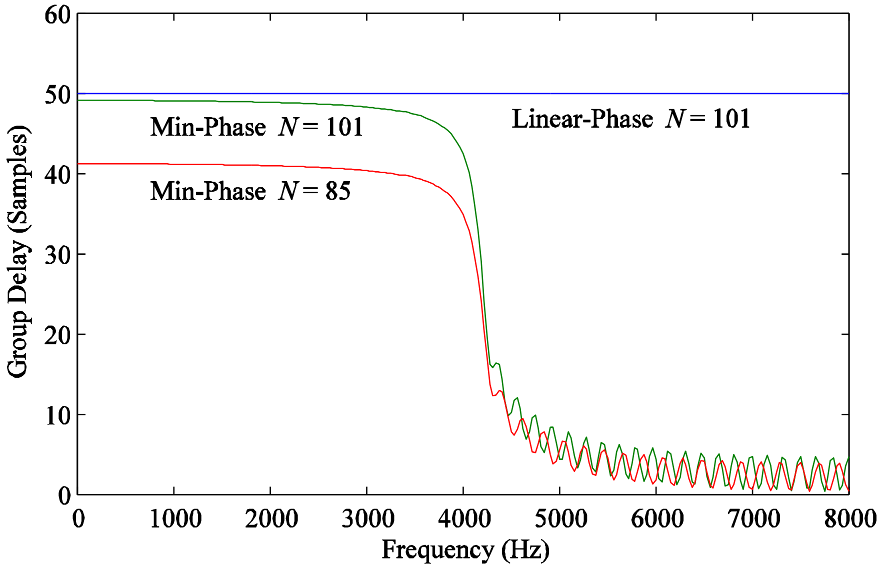

Fig. 6 Group delay for the linear-phase and minimum-phase filters.

## 3 Summary

This report has shown the benefits of frequency-weighting for linear-phase FIR design. The basic Remez-type algorithm for designing such filters, with the addition of constraints, can be put to use to design minimum-phase filters. By example, we have shown that minimum-phase filters have the obvious advantages of smaller delay, but also have better amplitude responses than linear-phase filters of the same length.

A method for mapping weights from the minimum-phase specification to the underlying linear-phase design was given. An efficient and effective approach for reconstructing the minimum-phase filter from its roots has been given.

## References

1. J. H. McClellan, T. W. Parks, and L. R. Rabiner, “A Computer Program for Designing Optimum FIR Linear Phase Digital filters”, *IEEE Trans. Audio, Electroacoustics*, vol. AU-21, pp. 506–526, Dec. 1973.
2. F. N. Fritsch and J. Butland, “A Method for Constructing Local Monotone Piecewise Cubic Interpolants", *SIAM J. Sci. Stat. Comput.*, vol. 5, pp. 300–304, June 1984.
3. R. P. Ramachandran and P. Kabal, “Minimax Design of Factorable Nyquist Filters for Data Transmission”, *Signal Processing*, vol. 18, no. 3, pp. 327–339, Nov. 1989.
4. F. Grenez, “Design of Linear or Minimum-Phase FIR Filters by Constrained Chebyshev Approximation”, *Signal Processing*, vol. 5, pp. 325–332, July 1983.
5. ITU-T Recommendation P.50, *Artificial Voices*, Sept. 1999.
6. A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd Edition, Prentice-Hall, 1998.
7. G. A. Sitton, C. S. Burrus, J. W. Fox, and S. Treitel, “Factoring Very-High-Degree Polynomials”, *IEEE Signal Processing Magazine*, vol. 20, no. 6, pp. 27-42, Nov. 2003.

# Appendix A Symmetric FIR Filters

Consider an FIR filter with $N$ real coefficients,

$$H(z) = \sum_{n=0}^{N-1} h[n] z^{-n}. \tag{5}$$

An even-symmetric or odd-symmetric filter satisfies the following equation,

$$h[n] = \pm h[N-1-n], \tag{6}$$

where the + sign applies to even-symmetry and the – sign applies to odd-symmetry. In $z$-transform notation,

$$H(z) = \pm z^{-(N-1)} H(z^{-1}). \tag{7}$$

## A.1 Root Factors

There are four filter types to consider: even- and odd-symmetric, and even and odd numbers of coefficients. Write $H(z)$ as follows,

$$H(z) = \frac{1}{2}[H(z) \pm z^{-(N-1)} H(z^{-1})]. \tag{8}$$

Consider $H(z)$ evaluated at $z = 1$,

$$H(1) = \frac{1}{2}[1 \pm 1^{-(N-1)}] H(1). \tag{9}$$

For the – sign (odd-symmetric), $H(1) = 0$, (any $N$). Now consider $H(z)$ evaluated at $z = -1$,

$$H(-1) = \frac{1}{2}[1 \pm (-1)^{-(N-1)}] H(-1). \tag{10}$$

For the + sign (even-symmetric), $H(-1) = 0$, if $N$ is even. For the – sign (odd-symmetric), $H(-1) = 0$, if $N$ is odd.

Noting the fixed roots found above, $H(z)$ can be factored as follows,

$$H(z) = Q(z) B(z), \tag{11}$$

where $Q(z)$ has only roots at $z = 1$ and/or $z = -1$. The term $B(z)$ is reduced in order, and as will be shown shortly, is even-symmetric with an odd number of coefficients. The results are summarized in the table below.

Table 1 Symmetric filter decomposition

| Type | $H(z)$ | $Q(z)$ | No. Coef. $B(z)$ |
|---|---|---|---|
| I | $N$ odd<br>even-symmetric | $1$ | $N$ |
| II | $N$ even<br>even-symmetric | $1+z^{-1}$ | $N-1$ |
| III | $N$ odd<br>odd-symmetric | $1-z^{-2}$ | $N-2$ |
| IV | $N$ even<br>odd-symmetric | $1-z^{-1}$ | $N-1$ |

The number of coefficients in $Q(z)$ is $N-M$, where $M$ is the number of coefficients in $B(z)$ and the symmetry in $Q(z)$ can be expressed as

$$Q(z) = \pm z^{-(N-M)} Q(z^{-1}). \tag{12}$$

The upper sign applies for $H(z)$ even-symmetric, while the lower sign applies for $H(z)$ odd-symmetric. Then

$$\begin{aligned} B(z) &= H(z)/Q(z) \\ &= \pm z^{-(N-1)} H(z^{-1})/Q(z) \\ &= \pm z^{-(N-1)} Q(z^{-1}) B(z^{-1})/Q(z) \\ &= \pm z^{-(N-1)} [\pm z^{N-M} Q(z)] B(z^{-1})/Q(z) \\ &= z^{-(M-1)} B(z^{-1}). \end{aligned} \tag{13}$$

In this equation, there are just two cases: $H(z)$ even-symmetric (use the plus signs) or $H(z)$ odd-symmetric (use the minus signs). This shows that $B(z)$ is always even-symmetric. It always has an odd number of coefficients as seen from Table 1.

## A.2 Frequency Response

The frequency response of a linear-phase filter can be expressed as

$$\begin{aligned} H(\omega) &= Q(\omega) B(\omega) \\ &= e^{-j\omega(N-M)/2} \tilde{Q}(\omega) e^{-j\omega(M-1)/2} B^0(\omega) \\ &= e^{-j\omega(N-1)/2} \tilde{Q}(\omega) B^0(\omega), \end{aligned} \tag{14}$$

where the response $B^0(\omega)$ can be expressed as

$$\begin{aligned} B^0(\omega) &= \sum_{n=0}^{M-1} b[n] e^{-j\omega(n-(M-1)/2)} \\ &= b[(M-1)/2] + 2\sum_{n=(M-1)/2+1}^{M-1} b[n]\cos(\omega(n-(M-1)/2)) \\ &= b[(M-1)/2] + 2\sum_{n=1}^{(M-1)/2} b[n+(M-1)/2]\cos(\omega n) \\ &= \sum_{n=0}^{(M-1)/2} \tilde{b}[n]\cos(\omega n), \end{aligned} \tag{15}$$

with the obvious definition of $\tilde{b}[n]$. This response is the *zero-phase response*. It is to be noted that $B^0(\omega)$ is a real function and can take on positive or negative values. The zero-phase response can be expressed as a Chebyshev polynomial in $x = \cos(\omega)$,

$$B^0(\omega) = \sum_{n=0}^{(M-1)/2} \alpha_n (\cos(\omega))^n. \tag{16}$$

The filter design program works with this transformed polynomial.

The fixed response $\tilde{Q}(\omega)$ is given in Table 2. As given, this is not a zero-phase response. The $j$ factor associated with the response can be taken out and lumped with the linear-phase term to give a generalized linear-phase term,

$$P(\omega) = e^{j\phi(\omega)}, \tag{17}$$

where the phase is

$$\phi(\omega) = \begin{cases} -\omega\dfrac{N-1}{2}, & \text{even-symmetric filter,} \\ -\omega\dfrac{N-1}{2} + \dfrac{\pi}{2}, & \text{odd-symmetric filter.} \end{cases} \tag{18}$$

Table 2 The response $\tilde{Q}(\omega)$

| Type | $H(z)$ | $\tilde{Q}(\omega)$ |
|---|---|---|
| I | $N$ odd<br>even-symmetric | $1$ |
| II | $N$ even<br>even-symmetric | $2\cos(\omega/2)$ |
| III | $N$ odd<br>odd-symmetric | $2j\sin(\omega)$ |
| IV | $N$ even<br>odd-symmetric | $2j\sin(\omega/2)$ |

## Appendix B Factoring a Linear-Phase Response

Consider an even-symmetrical filter with an odd number of coefficients. The symmetry condition can be expressed as

$$H(z) = z^{-(N_H-1)} H(z^{-1}). \tag{19}$$

This relation shows immediately that for every root $z_k$, there is another root at $1/z_k$. Furthermore, since we will deal with filters with real coefficients,

$$H(z) = H^*(z^*). \tag{20}$$

This means that roots come in complex conjugate pairs: $z_k$ and $z_k^*$. The two results together then mean that complex non-unit-circle zeros come in quads: $z_k$, $z_k^*$, $1/z_k$, and $1/z_k^*$, though not all of these need be distinct. There are a number of cases to be considered.

1. Complex, non-unit-circle zeros. These come in fours: conjugate pairs (two in the upper half plane, two in the lower half-plane) and which are at the same time reciprocals (two outside the unit-circle, two inside the unit-circle).
2. Complex, unit-circle zeros. These zeros come in complex conjugate pairs. The reciprocals coincide with the complex conjugate values.
3. Real valued (non-unit-magnitude) zeros. These roots come as reciprocal pairs.
4. Real (unit-magnitude) zeros. Roots at $z=1$ and $z=-1$ can appear as singles.

With this segregation of roots, a linear phase filter can be written as

$$H(z) = H_{\min}(z) H_{UC}(z) H_{\max}(x), \tag{21}$$

where $H_{\min}(z)$ is a minimum-phase filter (roots inside the circle), $H_{UC}(z)$ is a filter with roots on the unit-circle, and $H_{\max}(z)$ is a maximum-phase filter (roots outside the circle).

The unit-circle factors will be written as

$$H_{UC}(z) = \prod_{k=1}^{N_{UC}-1} (1 - z_k z^{-1}), \tag{22}$$

where $z_k = e^{j\theta_k}$. By grouping complex conjugate roots together to form second-order sections, each of those second-order sections takes on the linear-phase form,

$$H_{UCk} = 1 - 2\cos(\theta_k) z^{-1} + z^{-2}. \tag{23}$$

Any remaining real unit-circle zeros give terms of the linear-phase form $1 \pm z^{-1}$. Hence the filter $H_{UC}(z)$ is linear phase.

The gain will be split between the minimum-phase and maximum-phase filter. For every root $z_k$ inside the unit-circle which is part the minimum-phase filter, there will be another root $1/z_k$ outside the unit-circle which is part of the maximum-phase filter. We can form the minimum-phase filter and the maximum-phase filter as

$$\begin{aligned} H_{\min}(z) &= a \prod_{k=1}^{N_{\min}} (1 - z_k z^{-1}) \\ H_{\max}(z) &= a \prod_{k=1}^{N_{\max}} (z_k - z^{-1}). \end{aligned} \tag{24}$$

These filters have reciprocal roots, the same order ($N_{\max} = N_{\min}$) and are time reversals of each other,

$$H_{\max}(z) = z^{-(N_{\min}-1)} H_{\min}(z^{-1}). \tag{25}$$

The numbers of coefficients are related as follows

$$N_H = 2N_{\min} + N_{UC} - 1. \tag{26}$$

### B.1 Linear-Phase Filters with Double-Order Unit-Circle Zeros

Consideration an additional constraint: the unit-circle zeros must occur as double-order zeros. This is achieved within the design procedure by constraining the filter stopband to be nonnegative. With double-order zeros, the unit-circle zeros can be split, with one of each pair going to the minimum-phase filter and the other going to the maximum-phase filter. These modified filters will still be referred to as minimum-phase and maximum-phase even when they include the unit-circle zeros.

With these double-zeros allocated to the minimum-phase and maximum-phase filters,

$$\begin{aligned} H(z) &= H_{\min}(z) H_{\max}(z) \\ &= z^{-(N_{\min}-1)} H_{\min}^2(z) \\ &= z^{-(N_H-1)} H(z^{-1}), \end{aligned} \tag{27}$$

where

$$N_H = 2N_{\min} - 1. \tag{28}$$

The double-length filter $H(z)$ is always even-symmetric and of odd length. This means that we need not consider the other symmetries and lengths when designing factorable filters.

The double-length filter has correlation-like properties. The zero-phase part of the double-length filter has a positive frequency response and its largest coefficient is in the middle as can be seen from the following convolution sum,

$$\begin{aligned} h[n] &= \sum_{k=0}^{N_{\min}-1} h_{\min}[k] h_{\max}[n-k] \\ &= \sum_{k=0}^{N_{\min}-1} h_{\min}[k] h_{\min}[N_{\min} - 1 - n + k]. \end{aligned} \tag{29}$$

This attains its maximum for $n = N_{\min} - 1$,

$$h[N_{\min} - 1] = \sum_{k=0}^{N_{\min}-1} h_{\min}^2[k]. \tag{30}$$

### B.2 Segregating Roots

The procedure to find the minimum-phase factor of the double-length filter starts with finding the roots of the double-length filter. The Matlab routine `roots` is used for this step. The coefficients of the double-length filter are real and symmetric. The root-finding routine returns conjugate symmetric roots for real filter coefficients. However, the roots which should be reciprocals (due to the symmetry of the filter coefficients) are not exact reciprocals. In fact the deviation can in some cases be quite large.

The steps involved in classifying the roots are as follows.

1. Side information from the linear-phase design routine locates the positions of the double-roots on the unit-circle. These are extremal points in the design procedure. A tolerance of 1/10 the minimum spacing of these roots is established. Roots from the factored filter polynomial near these known points are removed. This removes the double-zeros on the unit-circle.
2. The remaining roots are sorted by magnitude and angle within similar magnitudes.
3. The roots are classified as to inside, unit-circle and outside roots. A zone is established around the unit-circle $(1 \pm 10^{-4})$ to allow categorization.

4. A check is made that non-unit-circle zeros occur in reciprocal pairs.[3]

5. Real non-unit-circle zeros are converged. Given two near reciprocal roots $z_I$ and $z_O$, calculate $r = z_I z_O$. Form $\hat{z}_I = z_I / \sqrt{r}$ and $\hat{z}_O = z_O / \sqrt{r}$.

6. Unit-circle zeros near $z = 1$ and $z = -1$ (imaginary part less than $10^{-4}$) are considered to be real zeros.

7. The single unit-circle roots at the extremal points are added back in.

### B.3 Forming the Minimum-Phase Filter

From the categorization of the roots, we can associate a set of roots corresponding to the minimum-phase filter. The minimum-phase filter must be formed from these roots. However, the forming of a polynomial from its root factors is a very ill-conditioned problem. The resulting polynomial can change dramatically depending on the order in which root factors are evaluated. The approach taken here is to evaluate the Discrete Fourier Transform of the root factors.[4] The motivation is that the frequency response can be calculated accurately from the roots and it is the frequency response that we want to preserve.

Factors involving pairs of complex conjugate roots are evaluated. The factors are of the form

$$\begin{aligned} G_{Cl}(z) &= (1 - z_l z^{-1})(1 - z_l^* z^{-1}) \\ &= z^{-1}(z^{-1} - 2\,\mathrm{Re}[z_l] + |\, z_l\,|^2\ z). \end{aligned} \tag{31}$$

The frequency response can then be evaluated as

$$G_{Cl}(\omega) = e^{-j\omega}((1 + |\, z_l\,|^2)\cos(\omega) + 2\,\mathrm{Re}[z_l] + j(1 - |\, z_l\,|^2)\sin(\omega)). \tag{32}$$

The $e^{j\omega}$ term represents a one-sample delay. The remaining term becomes purely real for $z_l = e^{j\theta}$. This property is simplifies part of the computation for filters, since generally a large number of the roots (those corresponding to the stopband region) lie on the unit-circle.

[3] An attempt was made to “converge” the reciprocal roots so that their product would be exactly unity. This procedure was not entirely successful. The resulting minimum-phase filters became decidedly non-equiripple. It was found best to just use the inside roots as they were returned by the root-finding routine.

[4] The Leja ordering procedure for ordering the roots has been promoted recently as a means to reduce the numerical problems [7]. Our experiments indicate that the DFT procedure is as good as the Leja ordering and more natural in the context of filter design.

For real roots, the roots are also paired,

$$G_{Rlm}(\omega) = e^{-j\omega}((1+z_l z_m)\cos(\omega) + z_l + z_m + j(1-z_l z_m)\sin(\omega)). \tag{33}$$

Since we have already sorted the roots, we can pair up small values with large values to help reduce the dynamic range of the root product term. Any remaining single real root is evaluated as

$$G_{Rl}(\omega) = 1 - e^{j\omega} z_l. \tag{34}$$

The delay terms $e^{-j\omega}$ are not explicitly evaluated — instead the accumulated integer-valued delay $D$ (in samples) is saved.

The frequency response is evaluated for a number of points that is a power of two greater than or equal to the number of coefficients $N_{\min}$. The frequency response is evaluated at uniformly spaced points on the unit-circle as the product of the root factors,

$$G[k] = \prod_l G_{Cl}(\omega_k) \prod_{p,m} G_{Rpm}(\omega_k)\, G_{Rq}(\omega_k) \tag{35}$$

where

$$\omega_k = \frac{2\pi k}{M}, \tag{36}$$

and $M$ is the number of points for the DFT. The frequency response is evaluated for $k$ from 0 and $M/2$. The remaining points are filled in using the complex conjugate relation,

$$G[k] = G^*[M-k], \qquad \text{for } M/2+1 \le k \le M-1. \tag{37}$$

The inverse DFT of the $M$ -point response is calculated as

$$g[n] = \frac{1}{N} \sum_{k=0}^{M-1} G[k] e^{j\frac{2\pi kn}{M}}. \tag{38}$$

A circular shift compensates for the delay,

$$\tilde{g}[n] = g[\mathrm{mod}(n-D, M)]. \tag{39}$$

Finally, the first $N_{\min}$ samples of the result are used as the impulse response of the minimum-phase filter. A check is made that the remaining terms (the terms with indices from $N_{\min}$ to $M-1$) of the inverse DFT are sufficiently small to be safely ignored.

The resulting polynomial is monic. A scaling factor is calculated so that Eq. (30) is satisfied

$$h_{\min}[n] = \pm a\tilde{g}[n] \qquad\qquad 0 \le n \le N_{\min} - 1, \tag{40}$$

where

$$a = \sqrt{\frac{h[N_{\min} - 1]}{\sum_{n=0}^{N_{\min}-1} \tilde{g}^2[n]}}. \tag{41}$$

There is still an ambiguity as to the sign to be associated with $h_{\min}[n]$. We evaluate the frequency response at $\omega = 0$ and at $\omega = \pi$, frequencies at which the frequency response is real,

$$H_{\min}(0) = \pm \sum_{n=0}^{N_{\min}-1} h_{\min}[n]; \qquad\qquad H_{\min}(\pi) = \pm \sum_{n=0}^{N_{\min}-1} (-1)^n h_{\min}[n]. \tag{42}$$

We choose the response value which has the largest magnitude and then adjust the sign of $h_{\min}[n]$ to give a positive value for that response.

# Appendix C Linear-Phase / Minimum-Phase Zero Locations

In this Appendix we investigate the relationship between the linear-phase filter and the minimum-phase filter in terms of frequency response and zero locations. For the purpose of illustration consider a minimum-phase filter with $N = 22$ coefficients (passband: $0 - 0.46\pi$, stopband: $0.54\pi - \pi$). We first design the double-length filter (with $2N - 1 = 43$ coefficients). The procedure discussed earlier in this document advocated the use of a design employing a non-negative constraint in the stopband. In the following, we show that for equiripple stopbands, the identical result can be obtained using a “lifting” procedure.

## C.1 Lifting Procedure for Equiripple Stopbands

If the weights are constant within the stopband, we can force the requisite double-order zeros on the unit-circle by a “lifting” procedure.

1. Design a linear-phase filter with equal stopband ripples. Let the resulting stopband ripple be $\delta_S$, i.e. the stopband oscillates between $+\delta_S$ and $-\delta_S$.
2. Lift the response by $\delta_S$ by adding this amount to the middle coefficient of the filter, coefficient $h[N-1]$. This lifts the frequency response of the zero-phase factor so that it is now non-negative. As the response is lifted, pairs of unit-circle zeros move toward each other to form double-zeros.
3. The lifting process also modifies the nominal passband value to $1 + \delta_S$. Compensate for the passband offset by scaling the filter by $1/(1 + \delta_S)$. The modified filter is then

$$\tilde{h}[n] = \frac{1}{1 + \delta_S}(h[n] + \delta_S \delta[N-1]). \qquad (43)$$

With weights chosen to match the passband ripple of the constrained design, this procedure gives identical results to the design with a stopband constrained to be non-negative. However, the constrained design is more flexible in that it does not require the stopband to be equiripple, i.e. the stopband can have a frequency-dependent weighting.

## C.2 Frequency Response of the Double-Length Filter

The frequency response of the zero-phase factor of a double-length filter so-designed is shown in Fig. 7. The ratio of the passband and stopband weights has been chosen so that the minimum-phase factor will have equal passband and stopband ripples. Fig. 8 shows the root plot. The roots that will go with the minimum-phase filter are shown as circles and those that will go with the maximum-phase filters are shown as squares.

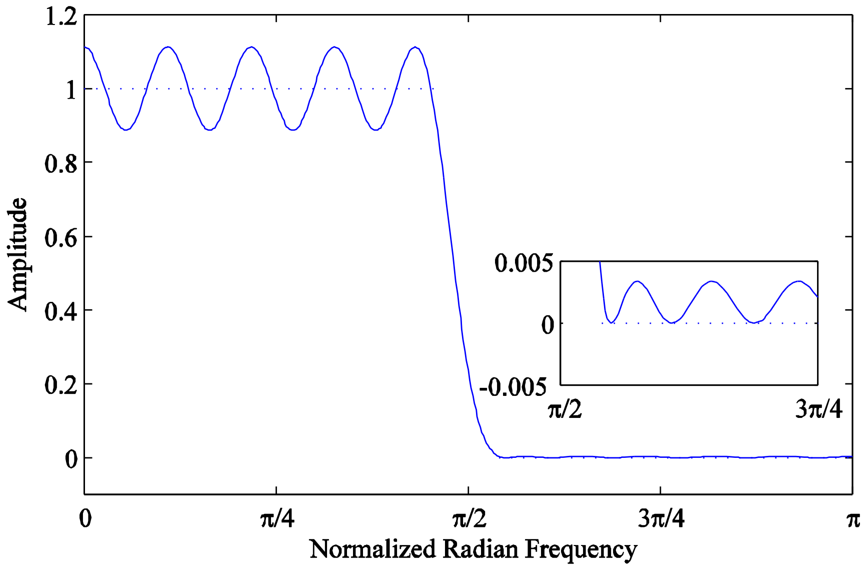

Fig. 7 Frequency response of the zero-phase factor of the double-length filter (length $2N-1$, where $N=22$).

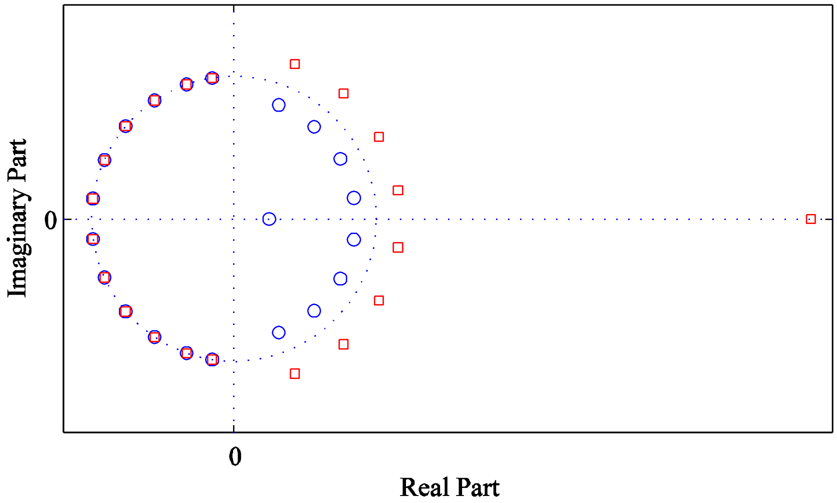

Fig. 8 Root plot for the double-length filter. Circles show the roots associated with the minimum-phase factor and squares show the roots associated with the maximum-phase factor.

## C.3 Frequency Response of the Minimum-Phase Filter

In the double-length filter, the stopband ripple is smaller than the ripple of the minimum-phase (or maximum-phase) factors, since the ripple is the product of the ripples of the minimum-phase and maximum-phase factors, see Fig. 9.[5] In the passband of the minimum-phase filter, the ripple of the double-length filter is larger than that of the factor filters.

Some facts to note: The passband of the minimum-phase filter has the same number of ripples as the double-length filter, i.e. twice that of a linear-phase filter that is the same length as the minimum-phase filter. However, the number of ripples in the stopband of the minimum-phase filter is half of the number in the double-length filter and is the same as that of a linear-phase filter of the same length as the minimum-phase filter.

## C.4 Weighting Relationships

Let us check on the relationship between the weights of the double-length filter and the minimum-phase filter as developed in Appendix D,

$$\frac{W_{SL}}{W_{PL}} = \frac{2V_M}{\delta_{SM}} \frac{W_{SM}}{W_{PM}}. \tag{44}$$

The ratio of stopband weight to passband weight used for the double-length filter was $W_{SL} / W_{PL} = 33.5$. The stopband ripple in the minimum-phase filter is $\delta_{SM} = 0.058$. The passband value of the minimum-phase filter is $V_M \approx 1$. The passband and stopband ripples of the minimum-phase filter are approximately equal, implying $W_{SM} / W_{SP} \approx 1$. These values check out reasonably well with the relationship above.

## C.5 Phase Response

The phase response for a linear-phase filter is linearly decreasing with frequency. For a minimum-phase filter, the phase is non-linear.

[5] This plot shows the zero-phase factor of the minimum phase filter. Since this filter is not linear-phase, the decomposition into a phase factor and a zero-phase factor requires phase unwrapping. Phase jumps of 2π are part of the normal phase wrapping. Phase jumps of π are associated with the change in sign of the zero-phase factor.

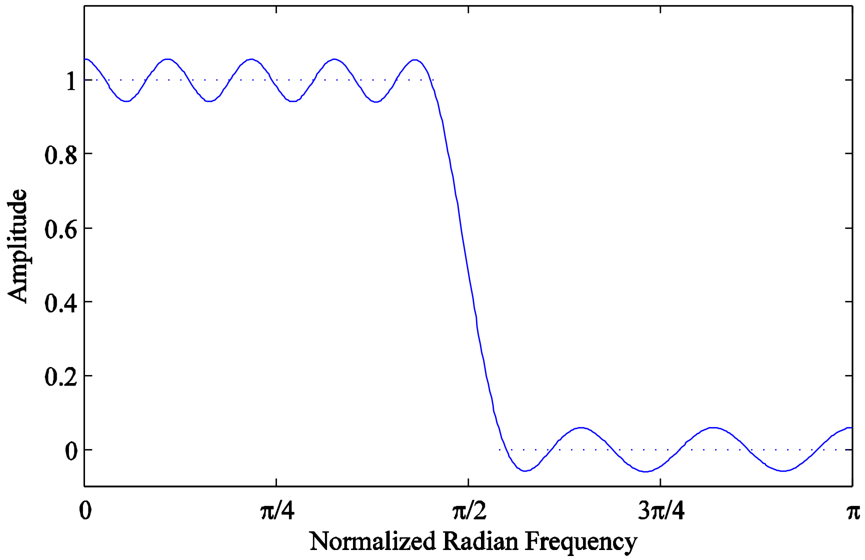

Fig. 9 Zero-phase factor of the minimum-phase filter, $N = 22$ .

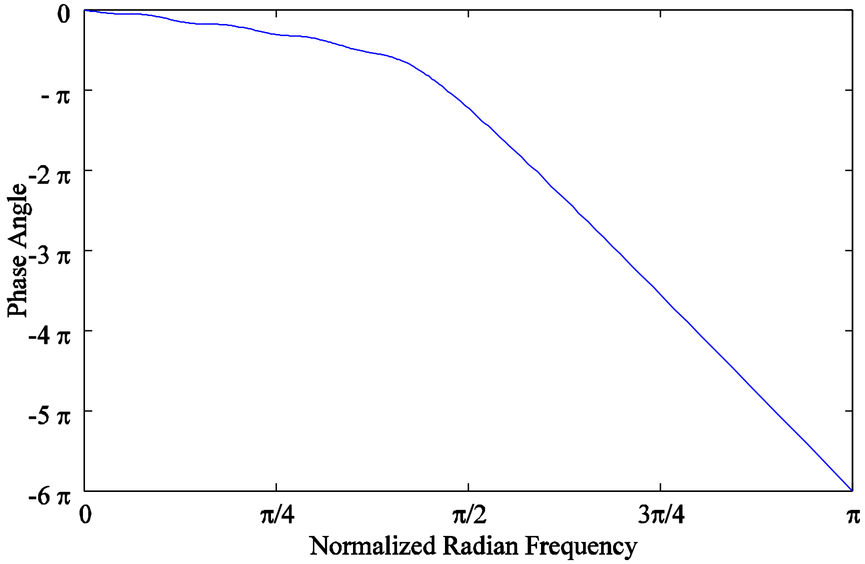

Fig. 10 Phase response for the minimum phase filter, $N = 22$ .

The maximum-phase factor can be expressed as product of the minimum-phase factor and an all-pass filter.

$$H_{\max}(\omega) = H_{\min}(\omega) H_{ap}(\omega). \tag{45}$$

The all-pass filter has poles superimposed on the zeros of the minimum-phase filter which lie inside the unit-circle. It has corresponding zeros at the reciprocal positions. The all-pass filter acts to “flip” the inside zeros to become outside zeros. The double-length linear-phase filter is the product of the minimum-phase factor and the maximum-phase factor.

$$H_{lp}(\omega) = H_{\min}(\omega) H_{\max}(\omega). \tag{46}$$

The phase responses can then be related as follows.

$$\begin{aligned} \phi_{\max}(\omega) &= \phi_{\min}(\omega) + \phi_{ap}(\omega) \\ \phi_{lp}(\omega) &= \phi_{\min}(\omega) + \phi_{\max}(\omega). \end{aligned} \tag{47}$$

Combining these we get

$$\phi_{\min}(\omega) = \frac{\phi_{lp}(\omega) - \phi_{ap}(\omega)}{2}. \tag{48}$$

The phase of the linear-phase filter $\phi_{lp}(\omega)$ is monotonic and it can be shown that the phase of an all-pass filter is monotonic (see for instance, [6]). The phase of the minimum-phase filter is the difference between these two phases.

The minimum-phase filter has zeros on the unit-circle in the stopband. The corresponding root factors form a linear-phase filter. In the passband, the zeros are not paired as reciprocal pairs – the phase contribution from those zeros is not linear-phase.

The unwrapped phase response that accompanies the zero-phase factor for the minimum-phase filter is shown in Fig. 10. One can note “wiggles” in the passband and the near-linear behaviour in the stopband.

### C.6 Group Delay Response

The group delay for a response is defined as the negative derivative of the phase response

$$\tau(\omega) = -\frac{d\varphi(\omega)}{d\omega}. \tag{49}$$

For linear-phase filters, the group delay is constant at $\tau(\omega) = (N-1)/2$ samples. For minimum-phase filters, the group delay is positive (due to the monotonicity of the phase response) and less than that for linear-phase filters, but is not constant.

The group delay for the minimum-phase filter is shown in Fig. 11. It was noted above that the phase response is nearly linear in the stopband. The fact that the group delay approaches a constant and does not exhibit oscillations in the stopband is consistent with this observation.

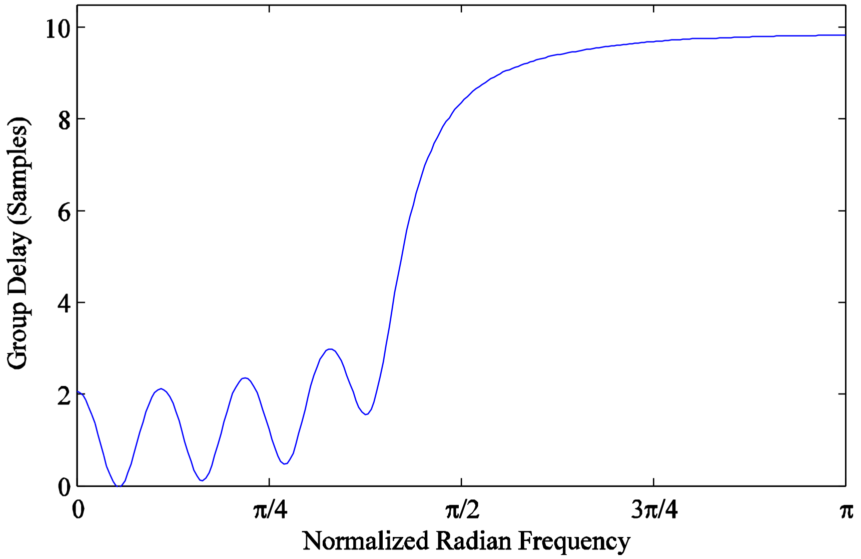

Fig. 11 Group delay for the minimum phase filter, $N = 22$ .

# Appendix D Weighting Factors for Minimum-Phase Filters

The minimum-phase filter design starts with a double-length linear-phase filter with double-order zeros on the unit-circle. This filter can be factored to give minimum-phase and maximum-phase filters. Each of the factored filters has the same magnitude response which is the square-root of the magnitude response of the double-length filter.

Let the minimum-phase filter have $N$ coefficients. Then the double-length filter has $2N-1$ coefficients. The double-length filter is linear phase with a constant group delay of $N-1$ samples. The delays of the minimum-phase and maximum-phase filters add up to this value, with the minimum-phase part having a group delay which is less than half of this value. The double-length filter will be equiripple. In the passband, the response will alternate about the desired value. In the stopband, the design constraints will force double-order zeros. This means that the response in the stopband will just touch zero. The corresponding minimum-phase filter will also be equiripple, however, the stopband ripple for the minimum-phase filter will alternate about zero.

## D.1 Desired Values and Deviations

Consider the design of a minimum phase filter with a single passband and a single stopband. Let the minimum-phase filter be designated with an “*M*” and the double-length linear-phase filter be designated with an “*L*”. For the passband, the desired value for the double-length filter is the square of the desired value for the minimum-phase filter. The passband of the minimum-phase filter will alternate between $V_M - \delta_{PM}$ and $V_M + \delta_{PM}$. The double-length filter will have passband values between the following limits

$$\begin{aligned} V_L - \delta_{PL} &= (V_M - \delta_{PM})^2 \\ V_L + \delta_{PL} &= (V_M + \delta_{PM})^2 . \end{aligned} \tag{50}$$

Then

$$\begin{aligned} V_L &= V_M^2 + \delta_{PM}^2 \\ \delta_{PL} &= 2 V_M \delta_{PM} . \end{aligned} \tag{51}$$

While we would like to take a specification of the desired value for the minimum-phase filter and translate that to a desired value for the double-length filter, this is not possible since the relationship involves the deviation $\delta_{PL}$ which is not known until after the design is completed. However, in the passband, if the deviation is small with respect to the desired value, the following approximation can be employed,

$$V_L \approx V_M^2 . \tag{52}$$

For the stopband, the double-length filter will have a one-sided ripple. The double-length filter will have stopband values from 0 to $\delta_{SM}^2$ , while the stopband of the minimum-phase filter will have values from $-\delta_{SM}$ to $+\delta_{SM}$ .

## D.2 Stopband and Passband Weighting

Let the passband and stopband weights for the minimum-phase filter be designated as $W_{PM}$ and $W_{SM}$ and the corresponding weights for the double-length filter be designated as $W_{PL}$ and $W_{SL}$ . Deviations are inversely proportional to weights,

$$\frac{\delta_{PM}}{\delta_{SM}} = \frac{W_{SM}}{W_{PM}} \quad \text{and} \quad \frac{\delta_{PL}}{\delta_{SL}} = \frac{W_{SL}}{W_{PL}} . \tag{53}$$

Using the previous results relating $\delta_{PL}$ to $\delta_{PM}$ ,

$$\begin{aligned} \frac{W_{SL}}{W_{PL}} &= \frac{2V_M \delta_{PM}}{\delta_{SM}^2} \\ &= \frac{2V_M}{\delta_{SM}} \frac{W_{SM}}{W_{PM}} . \end{aligned} \tag{54}$$

This shows that the weights of the double-length filter depend on the resultant stopband attenuation of the minimum-phase filter. It is not possible to specify, *a priori*, the ratio of the weights of the minimum-phase filter and use that ratio to calculate the ratio of the weights for the double-length filter. As a first approximation, one can estimate the stopband attenuation that will be achieved in the minimum-phase filter and use that to estimate the scaling factor for the weights. For instance if we want a stopband attenuation in the minimum-phase filter of 40 dB, then $V_M / \delta_{SM}$ will be $10^2$ and

$$\frac{W_{SL}}{W_{PL}} \approx 2 \times 10^2 \frac{W_{SM}}{W_{PM}} . \tag{55}$$

We skirt these issues by requiring the user to directly give the desired values and weights for the double-length filter. Given a desired weighting of the passband and stopband errors for the minimum-phase filter, Eq. (54) can be used to get a first estimate of the required weighting for the double-length filter. Equation (52) can be used as a first estimate of the desired value. The weights of the double-length filter and, if desired, the passband value of the double-length filter can be iterated for the final design.

## D.3 Frequency-Dependent Weighting

The design programs DFiltFIR and DFiltMPFIR allow for flexible frequency-dependent weighting by extrapolating between given weight values. The examples in the main text specify the weights at the edges of the stopband. The weights between these end points are found by linearly interpolating the weights ("interpolating in the linear domain"). The deviation in the stopband then varies inversely with the weights.

For the minimum-phase designs, the weights specified for the stopband of the double-length filter must compensate for the mapping between the deviations of the minimum-phase and double-length filters. We can make the stopband related values in Eq. (54) frequency-dependent,

$$\frac{W_{SL}(\omega)}{W_{PL}} = \frac{2V_M}{\delta_{SM}(\omega)} \frac{W_{SM}(\omega)}{W_{PM}}. \tag{56}$$

Consider two different frequencies $\omega_1$ and $\omega_2$. The ratios of the stopband weights (noting that deviations are inversely proportional to weights) are related as follows,

$$\begin{aligned} \frac{W_{SL}(\omega_1)}{W_{SL}(\omega_2)} &= \frac{\delta_{SM}(\omega_2)}{\delta_{SM}(\omega_1)} \frac{W_{SM}(\omega_1)}{W_{SM}(\omega_2)} \\ &= \left( \frac{W_{SM}(\omega_1)}{W_{SM}(\omega_2)} \right)^2 . \end{aligned} \tag{57}$$

For instance, to achieve a 10-fold reduction in deviation across the stopband (20 dB more attenuation) of the minimum-phase filter, the weights for the double-length filter must specify a 100-fold reduction in deviation. In addition, to achieve a linear change in weights in the minimum-phase filter, the weights for the double length filter must be interpolated in the square-root domain.[6]

For the passband, a similar analysis gives

[6] As well as options for interpolating in the linear and square-root domains, the minimum-phase design program also allows for interpolation in the log domain.

$$\frac{W_{PL}(\omega_1)}{W_{PL}(\omega_2)} = \frac{V_M(\omega_2)}{V_M(\omega_1)} \frac{W_{PM}(\omega_1)}{W_{PM}(\omega_2)}. \tag{58}$$

If the desired passband value $V_M(\omega)$ is the same at $\omega_1$ and $\omega_2$, the ratios of the weights at the two frequencies for the double-length filter and the minimum-phase filter are equal.

### D.4 Interpolation Domain

The filter design routines use monotonic cubic interpolation between given weight values. This interpolation occurs between triplets of reference points. If only two points are given, the interpolated values change linearly between the given points. Symbolically, the interpolated value at $x$ is determined from $P$ reference points,

$$y(x) = g_I(x;\{x_1 \cdots x_P\},\{y_1 \cdots y_P\}). \tag{59}$$

We can apply the interpolation in another domain by applying an invertible function, $f(y)$ to each reference ordinate point. Then interpolating in the $f(y)$ domain can be written as,

$$y(x) = f^{-1}(g_I(x;\{x_1 \cdots x_P\},\{f(y_1) \cdots f(y_P)\})). \tag{60}$$

The filter design programs offer $f(y) = y$ (linear domain), $f(y) = \sqrt{y}$ (square-root domain), and $f(y) = \log(y)$ (log domain). The square-root domain is useful for designing minimum-phase filters with interpolated stopband weights.

# Appendix E Matlab Code for the Examples

This appendix gives the Matlab code needed to generate the four filters in the main text.

```
% Highpass filter designs

SFreq = 16000;
fcS = 3850;
fcP = 4150;

% Linear-phase, constant stopband weighting
NCof = 101;
[B(1:2).Freq]   = deal ([0 fcS], [fcP SFreq/2]);
[B(1:2).Value]  = deal (0, 1);
[B(1:2).Weight] = deal (4.5, 1);
[B(1:2).WeightInt] = deal ('linear', 'linear');
h0 = DFiltFIR (NCof, B, 'bpf', SFreq);

% Linear-phase, frequency-weighted stopband
NCof = 101;
[B(1:2).Freq]   = deal ([0 fcS], [fcP SFreq/2]);
[B(1:2).Value]  = deal (0, 1);
[B(1:2).Weight] = deal (2.8*[10 1], 1);
[B(1:2).WeightInt] = deal ('linear', 'linear');
h1 = DFiltFIR (NCof, B, 'bpf', SFreq);

% Minimum-phase, frequency-weighted stopband
NCof = 101;
[B(1:2).Freq]   = deal ([0 fcS], [fcP SFreq/2]);
[B(1:2).Value]  = deal (0, 1);
[B(1:2).Weight] = deal (12000*[10 1].^2, 1);
[B(1:2).WeightInt] = deal ('sqrt', 'linear');
h2 = DFiltMPFIR (NCof, B, SFreq);

% Minimum-phase, frequency-weighted stopband
NCof = 85;
[B(1:2).Freq]   = deal ([0 fcS], [fcP SFreq/2]);
[B(1:2).Value]  = deal (0, 1);
[B(1:2).Weight] = deal (900*[10 1].^2, 1);
[B(1:2).WeightInt] = deal ('sqrt', 'linear');
h3 = DFiltMPFIR (NCof, B, SFreq);
```