Filter design using MATLAB

Article · January 1999

3 authors:

Per Bertil Löwenborg
Signal Procession Devices

Oscar Gustafsson
Linköping University

Lars Wanhammar
Linköping University

Filter Design Using MATLAB

Per Löwenborg, Oscar Gustafsson, and Lars Wanhammar

Department of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden.
E-mail: {perl, oscarg, larsw}@isy.liu.se

Abstract

This paper discusses the design of analog and digital filters using MATLAB from Mathworks Inc. Advantages and disadvantages with the existing Signal Processing Toolbox are discussed and a set of new or improved functions are proposed. Included among the new functions is a pole-placer, used to design filters with arbitrary piecewise constant stopbands. Together these new functions will provide a more competent and robust filter design toolbox.

1 Introduction

Synthesizing analog and digital filters meeting non-standard requirements is a complex task that include non-trivial mathematics that necessitates extensive use of computer software. One of the leading software packages for numerical calculations is MATLAB [1][1] from Mathworks Inc. One of the "toolboxes" is the Signal Processing Toolbox [2], which includes a set of filter design functions, including some very basic functionality for designing standard filter approximations. However, most auxiliary functions are based on the rational functions in polynomial form, which has very poor computational properties. To alleviate this shortcoming we introduce the corresponding functions based on using the poles and zeros directly or on so-called transformed variables. The latter not only improves the numerical accuracy, but also simplifies the synthesis problem by separating the stopband and passband into two almost independent problems. This and other modifications improves the accuracy and robustness significantly.

In this paper we investigate the suitability of using MATLAB and the Signal Processing Toolbox for filter design and discusses possible improvements and extensions. For simplicity we will refer to the Signal Processing Toolbox as the toolbox. Of special importance is the pole-placer functions that is used to synthesize filters with equiripple or maximally flat passband and arbitrary piecewise constant stopbands. The new functions form a filter design toolbox suitable for high-performance analog and digital filter synthesization problems. All functions are described in the appendix of this paper.

1. MATLAB is a registered trademark of Mathworks Inc.

2 Pros and Cons

The main advantage of using an already existing software package such as MATLAB is the built-in support for basic functions, like for example, matrix operations and graphics. This enables rapid development of new design routines.

2.1 Polynomials vs. Roots

A linear, time-invariant filter can be described in a number of different ways. Using the Laplace-transform, the transfer function for a filter can be written

$$H(s) = \frac{b(s)}{a(s)} = \frac{\sum_i b_i s^i}{\sum_j a_j s^j} = g\frac{\prod_i (s - z_i)}{\prod_j (s - p_j)} \qquad (1)$$

Here we see that we can represent the transfer function in two different ways, either with the coefficients of the polynomials (the a_i's and b_j's) or the zeros and poles of the polynomials (the z_i's and p_j's). These representations will have different computational properties. For higher order or narrow transition band filters the difference in accuracy is significant. The use of such high performance analog filters are mainly as reference filters in the design of discrete-time and digital filters. Figure 1 shows the attenuation for a 19th-order Cauer filter plotted in MATLAB using both the standard function (*freqs.m*) and our new one based on pole-zero representation. The requirement on passband ripple and transition band are chosen very high in order to clearly show the difference in accuracy between the two approaches. The specifications for the filter is

$$A_{max} = 0.001 \text{ dB} \qquad f_c = 1 \text{ Hz}$$
$$A_{min} = 40 \text{ dB} \qquad f_s = 1.001 \text{ Hz}$$

New functions for the transformations between lowpass and highpass, bandpass, and bandstop filters respectively using the pole-zero representation have also been implemented.

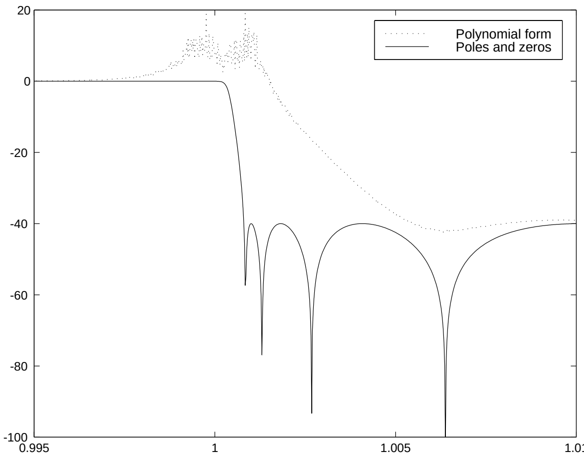

Figure 1: *Comparison of magnitude response of a 19th-order Cauer filter calculated using the Signal Processing Toolbox and a new function.*

2.2 Design Margin

When a filter is designed the order of the filter must be an integer number. However, the theoretical filter order is in most cases not an integer. The difference between the chosen order and the theoretical is called the design margin and can be used in different ways to improve filter performance and reduce the implementation cost. Hence it is important to know how the design margins are handled as it may be used differently depending on the application. Thus, is it also important to know where the filter design program puts the design margin. Table 1 shows how the MATLAB Signal Processing Toolbox uses the design margin for the standard filter approximations. This is, however, not necessarily the way the designer would like to utilize the available design margin.

Table 1: *Use of design margin for standard filter approximation lowpass filters in MATLABs Signal Processing Toolbox.*

Approximation	Design margin utilization
Butterworth	Decreased passband attenuation
Chebyshev I	Increased stopband attenuation or decreased transition band
Chebyshev II	Decreased transition band
Cauer (elliptic)	Decreased transition band

By using the new functions the user has full control to allocate the design margins. New functions giving the limits of the constraints for a given order have been implemented and give better understanding of possible utilization of the design margin.

3 The Pole-placer

The original pole-placer algorithm was introduced by B.R. Smith and G.C. Temes and is described in [5]. It uses transformed variables [4], which separates the filter passband and stopband by mapping different parts of the $j\omega$-axis in the s-plane to the real- and imaginary axis in the Z-plane. The relation between the Z-plane and the s-plane is[1]

$$Z^2 = \frac{s^2 + \omega_B^2}{s^2 + \omega_A^2} = \frac{\omega_B^2 - \omega^2}{\omega_A^2 - \omega^2} \qquad (2)$$

where ω_A and ω_B are the lower and upper limits of the passband. Thus is $\omega_A = 0$ for a lowpass filter and Eq. (2) becomes

$$Z^2 = 1 + \frac{\omega_B^2}{s^2} = 1 - \frac{\omega_B^2}{\omega^2} \qquad (3)$$

By using transformed variables, the numerical accuracy is improved in the passband due to the fact that the transformation maps clustered zeros further apart [4]. This is particularly useful since the poles in the passband tend to cluster close to the transition region, where also the sensitivity is the worst. This can be seen in Fig. 1. In the stopband the zero locations are compacted slightly, but they are still sufficiently separated, not to cause any significant reduction in numerical accuracy.

For a normal filter specification, using N decimal digits, a filter can be designed with sufficient accuracy only for filter orders up to about N, performing the synthesis in the s-plane as is the case for textbook based design routines, like MATLAB Signal Processing Toolbox. Using the transformed variable, filters with filter orders up to about $3N$ can easily be synthesized [6].

The pole-placer algorithm works with attenuation poles, i.e., transmission zeros, and iterates the position of the attenuation poles until an optimal filter is achieved. The optimality criterion is that the minimum distance between each arc of the attenuation function and the attenuation constraint should be the same for all arcs, as can be seen in Fig. 2.

Starting with an initial set of attenuation poles, the position of the minimum attenuation of each arc are determined. Next, for each arc the minimum distance between the attenuation function and the attenuation constraint is found and the frequency at which it occurs. If the difference between the largest and the smallest difference is small enough the iteration has converged and we have an optimal set of attenuation poles in the Z-plane. If there is no convergence we iteratively compute new values for the attenuation poles, while making sure that these are within

1. A slightly more accurate definition is [6]

$$Z^2 = \ln\left(\frac{s^2 + \omega_B^2}{s^2 + \omega_A^2}\right)$$

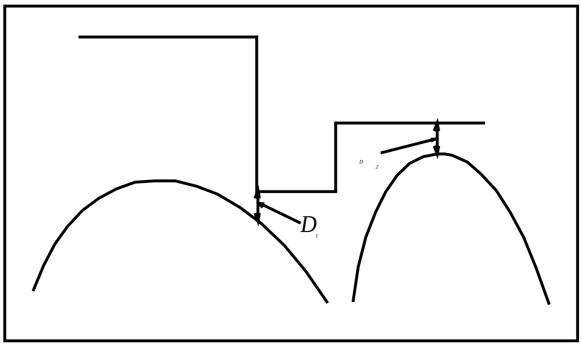

Figure 2: *Minimum distances to specification for the attenuation arcs in the stopband.*

the stopband region.

When the attenuation poles have been determined the reflection zeros are calculated. Considering a doubly resistively terminated lossless two-port, which is optimal from sensitivity point of view, the reflection zeros are the frequencies at which maximum power is transferred to the load. For the equiripple and maximally flat passband filter cases, the reflection zeros are uniquely determined by the transmission zeros found in the iterative pole-placing procedure. Through the Feldkellers equation [3, 6], the natural frequencies (the poles) can be found using the reflection and transmission zeros and by a scaling of the magnitude response. The zeros and poles can the be found by transforming back to the s-plane. Also component values for ladder networks can be found directly using the *Z*-plane variables [4].

This far, pole-placer functions has been developed for MATLAB for the lowpass, highpass, and bandpass filter cases, having either maximally flat or equiripple passbands. A pole-placer algorithm for equiripple stopband filters is developed in [7].

Since analytic expressions of the attenuation and its partial derivatives with respect to the attenuation pole frequencies are available, the convergence time for the implemented pole-placing algorithms is very short.

4 Examples

4.1 Example of Design Margin

Assume that a lowpass Chebyshev I filter with the following specifications is to be designed

$$A_{max} = 1 \text{ dB} \qquad f_c = 1 \text{ Hz}$$
$$A_{min} = 25 \text{ dB} \qquad f_s = 3 \text{ Hz} \tag{4}$$

This filter will have an order of three. Using the standard function (*cheb1ap.m*) with the standard arguments we will get the poles and zeros of the filter plotted with the solid line in Fig. 3. All of the design margin is allocated to increase the stopband attenuation or decrease the transition band. By using the new design margin function we can get the extreme values that one of the specification components may be changed to. These values are shown in Eq. (5).

$$A_{max} = 0.1375 \text{ dB} \qquad f_c = 1.376 \text{ Hz}$$
$$A_{min} = 34.046 \text{ dB} \qquad f_s = 2.1798 \text{ Hz} \tag{5}$$

The A_{min} and f_s values represents the original poles and zeros, i.e., the solid line in Fig. 3, while allocating the design margin to decrease the passband ripple corresponds to the dashed line in Fig. 3. Finally, it is possible to allocate the design margin for a larger passband, as shown by the dotted line in Fig. 3. All these are valid third-order Chebyshev I filters which satisfy to the requirements in Eq. (4), but have used the design margin differently.

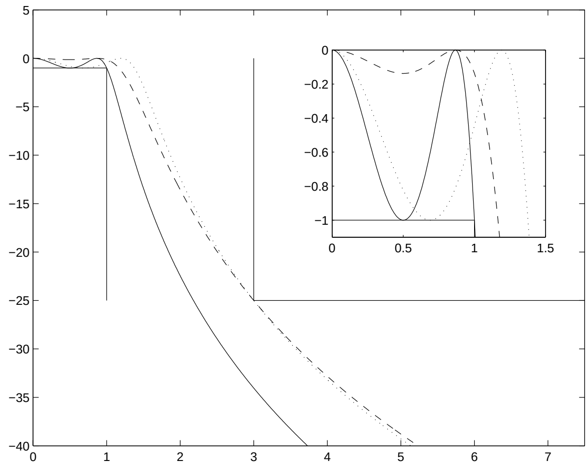

Figure 3: *Magnitude response of three third-order Chebyshev I filters with different utilization of the design margin.*

4.2 Pole-placer – Example 1

This example illustrates the increased numerical accuracy using the transformed variable in the pole-placer and the increased accuracy achieved by using the transformed variable. Consider the following hypothetical filter specification[1] for a lowpass Cauer filter:

$$A_{max} = 0.00001 \text{ dB} \qquad f_c = 1 \text{ Hz}$$
$$A_{min} = 300 \text{ dB} \qquad f_s = 1.1 \text{ Hz} \tag{6}$$

The resulting filter order is 39. Figure 4 shows the resulting magnitude response calculated with the new frequency response function. A deviation from equiripple behaviour near the passband edge can be seen and is caused by the clustering of poles there. An attempt to use the *freqs.m* function in MATLAB Signal Processing Toolbox for calculating the magnitude response yields a large deviation from the magnitude response shown in Fig. 4.

Further it can be seen that the pole-placer has sufficient accuracy to synthesize as high filter orders as 39 or more.

1. The specification is chosen extremely stringent in order to clearly show the difference in numerical accuracy

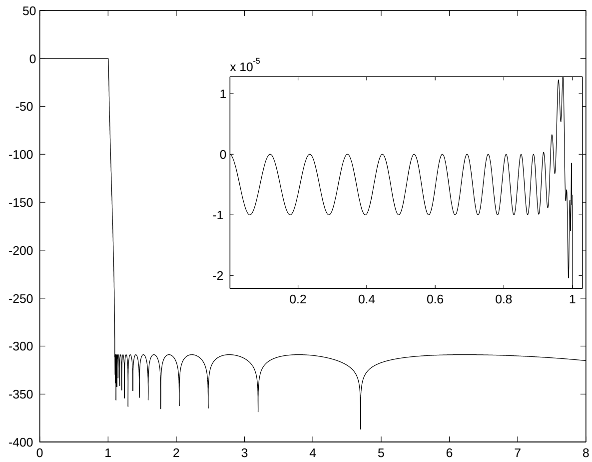

Figure 4: *Magnitude response of 39th-order Cauer filter designed with the pole-placer.*

The highest possible filter order for a Cauer lowpass filter using MATLAB standard functions is in this case about 18, which results in a stopband attenuation of 106 dB, satisfying the rest of the specification in Eq. (6).

4.3 Pole-placer – Example 2

Considering the following specification for a maximally flat passband lowpass filter with 0.1 dB passband ripple and a cut-off frequency of 1 Hz. The stopband attenuation is specified to 40 dB between 2 and 3 Hz and 25 dB above 3 Hz. The resulting filter order using the pole-placer is 5. The magnitude response is shown in Fig. 5. Using a standard approximation Chebyshev II filter in MATLAB Signal Processing Toolbox we get the filter which magnitude response is shown in Fig. 6. In order to meet the specification the resulting filter order is 6. Thus, by using the pole-placer one can achieve lower filter orders. Also a better utilization of the design margin is possible as can be seen from the transition regions in Fig. 5 and Fig. 6.

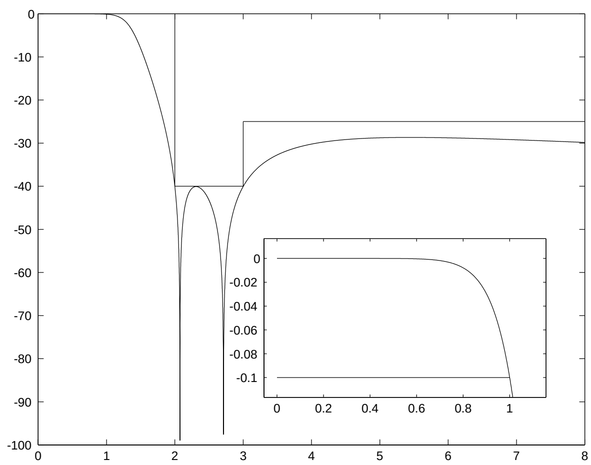

Figure 5: *Magnitude response of lowpass filter with maximally flat passband designed with the pole-placer.*

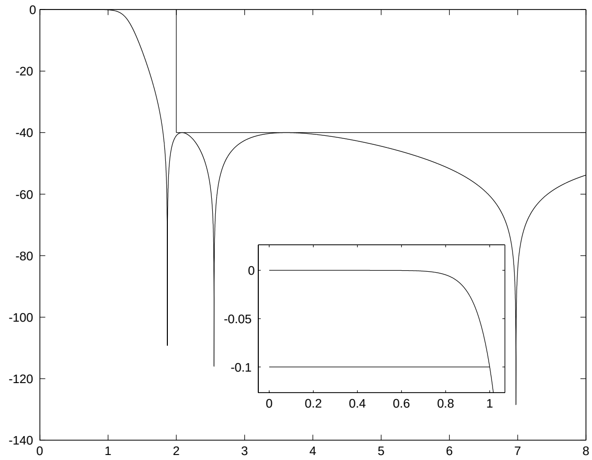

Figure 6: *Standard Chebyshev II lowpass filter magnitude response.*

4.4 Pole-placer – Example 3

The textbook way of designing a bandpass filter is to first design a lowpass prototype filter and the transform it to a bandpass filter using a lowpass to bandpass transformation. This transformation restricts the bandpass filter to be geometrically symmetrical and with only one stopband attenuation. Using the pole-placer instead, any equiripple or maximally flat passband, bandpass filter can be designed with multiple piecewise constant stopband requirements, not only the geometrically symmetrical ones. The filter shown in Fig. 7 was designed using the pole-placer for the specification: $A_{max} = 0.01$ dB, lower stopband attenuation 50 dB and stopband frequency 1 Hz, upper stopband attenuation 50 dB and 70 dB and stopband frequency 5 and 6 Hz respectively. The required filter order was 8. The passband frequencies were 2 and 3 Hz respectively. Note the non-geometrically symmetric specification.

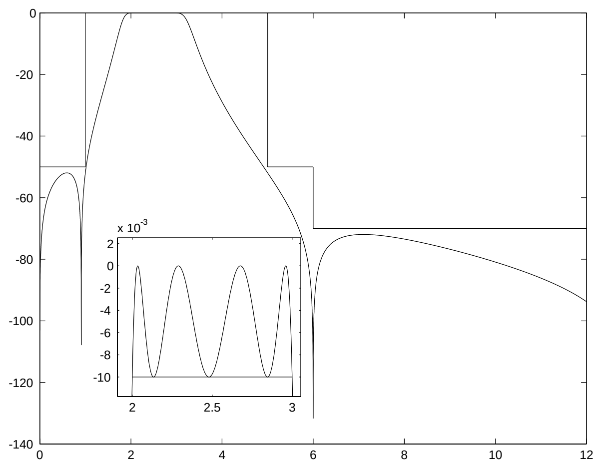

Figure 7: *Magnitude response of an equiripple passband, bandpass filter designed with the pole-placer.*

5 Future Work

LC filters are often used as reference filters for designing advanced active, mechanical, and digital filters. We will therefore develop a synthesis toolbox for *LC* filters based on the pole-placer that will exploit the superior accuracy of the transformed variable.

6 Conclusion

Using the pole-zero representation is advantageous in most cases. We have developed MATLAB routines for computing the frequency, magnitude, phase, and group delay responses based on a pole-zero representation, that have superior numerical accuracy compared to the routines supported in the MATLAB Signal Processing Toolbox.

We have also developed a pole-placer that has superior numerical accuracy and allows the design of non-standard analog and digital filters with piecewise constant stopband requirements and maximally flat or equiripple passbands.

References

[1] http://www.mathworks.com/. Dec. 1998.

[2] *Matlab, Signal Processing Toolbox User's Guide*. The MathWorks, Inc., 1996.

[3] R.W. Daniels, *Approximation Methods for Electronic Filter Design*, McGraw-Hill Book Co., New York, USA, 1974.

[4] H.J. Orchard and G.C. Temes. "Filter Design Using Transformed Variables". *IEEE Trans. Circuit Theory*, Vol. 15, pp. 385-408, Dec. 1968.

[5] B.R. Smith and G.C. Temes. "An Iterative Approximation Procedure for Automatic Filter Design". *IEEE Trans. Circuit Theory*, Vol. 12, pp. 107-112, Mar. 1965.

[6] A.S.Sedra and P.O. Bracket. *Filter Theory and Design: Active and Passive*. Matrix Publishers, Inc., Portland, Oregon, USA, 1978.

[7] J. Teng, J.K. Fidler, and L. Zheng, "Pole Placement Algorithm for Equiripple Bandstop Filter Design". *IEE Proc. Circuits, Devices and Syst.*, Vol. 141, No. 4, Aug. 1994.

Appendix - Toolbox Functions

lpeq_pp.m	Lowpass pole-placer, equiripple passband.
hpeq_pp.m	Highpass pole-placer, maximally fast passband.
bpeq_pp.m	Bandpass pole-placer, equiripple passband.
lpmf_pp.m	Lowpass pole-placer, maximally flat passband.
hpmf_pp.m	Highpass pole-placer, maximally flat passband.
bpmf_pp.m	Bandpass pole-placer, maximally flat passband.
ppdemo.m	Demonstration of the pole-placer functions.
qsgd.m	Analog filter groupdelay for quadrantally symmetrical zeros [3].
groupdelay.m	Groupdelay for general analog filters (only for digital filters in the Signal Processing Toolbox),
buttmargin.m	Butterworth design margins for a given specification.
cheb1margin.m	Chebyshev I design margins for a given specification.
cheb2margin.m	Chebyshev II design margins for a given specification.
ellipmargin.m	Cauer design margins for a given specification.
margindemo.m	Demonstration of possible utilization of the design margin.
zp2hp.m	Lowpass to highpass transformation for poles and zeros.
zp2bp.m	Lowpass to bandpass transformation for poles and zeros.
zp2bs.m	Lowpass to bandstop transformation for poles and zeros.
zp2frs.m	Frequency response based on poles and zeros.