A LADDER PROTOTYPE SYNTHESIS ALGORITHM

R. D. Kübler
B. M. Wilamowski
Department of Electrical Engineering
University of Wyoming
Laramie, WY 82070

Abstract: Ladder filter structures are important for filter designs with modern circuit elements. This paper presents a modern algorithm for ladder filter synthesis. This method is general, and only a few differences in the continued fraction expansion are all that separates a Butterworth synthesis from a Cauer-Elliptic synthesis. The generality of this approach also allows the synthesis of transfer functions designed by non-traditional methods. In addition, the algorithm generates many possible circuit implementations for transfer functions with zeros. Finally, a software package LADDER implements the algorithm. In addition to ladder synthesis, LADDER includes front end transfer function approximation algorithms and post-synthesis analysis of ladder circuits.

INTRODUCTION

Modern filter designs such as Switched Capacitor filters or Switched Current filters use LC ladder prototypes [1,2]. Using these prototypes for Girling-Good type active filters results in smaller sensitivity than conventional cascade filters [3]. Therefore, the synthesis of LC ladder filters is of current attention. The purpose of this work is to develop and implement a general algorithm for ladder filter synthesis.

Currently, when designs require a ladder prototype, a designer completes the task with the aid of tables [4]. Because of their double and triple entry formats, these tables are difficult to use. The complex numerical methods involved in generating tables are far from general. Transfer functions with zeros require very complex and filter specific algorithms. The most common approach, Cauer's algorithm[5], takes advantage of a particular filter's properties, so each filter approximation technique needs its own specific algorithm. This lack of generality makes it an impractical choice for a general filter synthesis algorithm. For transfer functions with zeros, Cauer's approach produces only one circuit implementation, where many are possible.

The synthesis procedure developed herein, allows the generation of not just one but many ladder circuit representations for a given transfer function with zeros. The algorithm is general; it works for all types of transfer function approximations from Butterworth through Cauer, and even non-traditional transfer function designs.

PRINCIPLE OF THE ALGORITHM

For transfer functions without zeros, such as Butterworth, Chebyshev, and Bessel-Thompson type filters, determining element values involves the classical continued fraction expansion of an input impedance [6].

![Image 1](image.png)

Figure 1. One section of a lowpass ladder network with zeros.

The more challenging synthesis is for transfer functions with zeros, such as Cauer and Inverse Chebyshev filter types. Using a general approach, involving the calculation of the input impedance using an auxiliary function, the function $A(z)$ is easy to find [3]. Some problems arise when obtaining $A(s)$ from $A(j\omega)^2$. This separation uses two methods: 1) find the roots and eliminate those in the right half plane; 2) solve a system of nonlinear equations. [7]

The first problem, in solving the synthesis of a given impedance with transmission zeros, is the determination and removal of the shunt capacitance, $C_s$. One method is discussed in [8]. That method, however, gives only one choice for the shunt capacitor. Using the algorithm developed herein, it is possible to choose from as many capacitors as there are zeros remaining to be removed. Because any remaining zero can be removed at any time, the total number of possible solutions to the synthesis problem is $n_z^n$, where $n_z$ is the number of conjugate zeros.

Determination of the capacitor value to remove is a complex procedure. For a given impedance function, $Z_A(s)$ in (1), the removal of the shunt capacitor, $C_s$, will lead to equation (2), where the values for the $k_i$ are given in (3).

$$Z_A(s) = \frac{N_A}{D_A} = \frac{a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \ldots + a_1s + a_0}{b_{m-1}s^{m-1} + b_{m-2}s^{m-2} + \ldots + b_1s + b_0} \tag{1}$$

$$Z_B(s) = \frac{N_B}{D_B} = \frac{a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \ldots + a_1s + a_0}{b_{m-1}s^{m-1} + k_1s^{m-2} + k_2s^{m-3} + \ldots + k_{m-1}s + k_m} \tag{2}$$

$$k_i = b_i - C_s a_i \text{ for } (1 \leq i \leq n) \quad (k_0 = b_0) \tag{3}$$

To remove the $L_s C_s$ resonant circuit, the denominator of $Z_B(s)$ should be divisible by $(L_s C_s)^{n_z^2 + 1}$ with a remainder of 0. Figure 2 illustrates the division of the denominator of $Z_B(s)$ by $z^{2n_z^2 + 1}$, where $z^2 = (L_s C_s)^{1/2}$. Equations for $R_1$ and $R_0$ are given by (4) and (5), respectively. In order for the remainder to be 0, the numerator of both of these coefficients must be 0.

$$k_{m-3}z^2 + k_{m-2}z^3 + \ldots + k_2z^2 + k_0 \tag{4}$$

$$k_{m-1}z + \frac{k_{m-2}z^2}{z^2} + \ldots + k_2z^2 + k_0 \tag{5}$$

Figure 2. Long division of the denominator of $Z_B(s)$ by $(z^2)^{2n_z^2 + 1}$.
Setting the numerators of (4) and (5) equal to zero, substituting (3) in for $k_i$, and solving for $C_5$ yields equations (6) and (7).

$$C_5 = \frac{b_1 z^{(N-1)} \cdot b_2 z^{(N-3)} + b_2 z^{(N-5)} \cdots}{a_2 z^{(N-1)} \cdot a_2 z^{(N-3)} + a_2 z^{(N-5)} \cdots}$$

(6)

$$C_5 = \frac{-b_1 z^{(N-1)} \cdot b_2 z^{(N-3)} + b_2 z^{(N-5)} \cdots}{a_2 z^{(N-1)} \cdot a_2 z^{(N-3)} + a_2 z^{(N-5)} \cdots}$$

(7)

Concurrent solution of equations (6) and (7) yields the required shunt capacitor value.

After removing the shunt capacitor, the next step becomes determining the resonant capacitor and inductor values. Using the model shown in Figure 1, the impedance function at $B$, $Z_B$, is the sum of the resonant impedance and the impedance function at $C$, $Z_C$, as shown in (8). After finding a common denominator, equation (9) shows a relation between the denominators of $Z_C$ and $Z_B$. Next, substituting (9) into (8), and solving for $L_R$ yields (10). In (10), the knowns are $N_B$, $D_C$, and $z$, and the unknown is $N_C$. Evaluating (10) at $s=jz^2$ eliminates the unknown, allowing the inductor to be written in terms of all knowns (11).

$$Z_B = \frac{N_B}{D_B} = \frac{L_R z}{L_R C_R z^2 + 1} + \frac{N_C}{D_C} = \frac{L_R z}{z^2 z^2 + 1}$$

(8)

$$D_B = D_C(z^2 z^2 + 1)$$

(9)

$$L_R = \frac{(z^2 z^2 + 1)N_B}{D_B z^2} - \frac{(z^2 z^2 + 1)N_C}{D_C z^2} = \frac{N_B - (z^2 z^2 + 1)N_C}{D_C z^2}$$

(10)

$$L_R = \frac{N_B}{D_C z^2} \bigg|_{s \rightarrow jz^2} = z_B(s) = \frac{(z^2 z^2 + 1)}{s} \bigg|_{s \rightarrow jz^2}$$

(11)

After determining the inductor value, subtracting the resonant impedance from the impedance function removes the complete parallel LC circuit.

The process of removing the shunt capacitor and resonant LC circuit continues until all of the resonant circuits have been removed. At this point, the remaining shunt capacitor, inductor (in the case of an even design), and output resistor are found using classical continued fraction.

**Algorithm Implementation**

A new software package LADDER implements the algorithm developed herein. LADDER not only does ladder network synthesis but also generates transfer functions and completes the design with post-synthesis analysis. To demonstrate the algorithm, a 7th order lowpass Cauer filter, with $\omega_p = 5$, $\omega_s = 100$, $\omega_p = 1.0$, and $\omega_s = 2.0$, is synthesized using LADDER.

Using modern menu driven software, LADDER guides the user through the transfer function design. LADDER offers brick wall specifications, with an optional for manual or automatic order specifications, for traditional functions such as Butterworth, Chebyshev, Inverse Chebyshev, Cauer, and Bessel-Thompson. Additionally, LADDER allows the design of transfer functions transfer functions using interactive pole/zero placement.

Synthesis of transfer functions with equal order numerator and denominator polynomials requires an additional transformation. This transformation, (14), transforms the largest conjugate zero pair, $\omega_{max}$, to infinity, thus reducing the order of the numerator by 2, while still maintaining the same basic magnitude response. The normalization frequency, $\omega_N$, maps to the same frequency on both the $\omega$-axis and the $\Omega$-axis.

$$\omega^2 = \frac{\omega^2 - \omega^2_{max}}{\omega^2 - \omega^2_{max}}$$

(14)

Another transformation is for filters without a maximum at zero, for example even order Cauer and Chebyshev filters. This transformation moves the frequency of the first maximum, $\omega_{max}$ in (15), to zero and allows the design of even order Cauer and Chebyshev filters with equal terminating resistances. The importance of this transformation comes from the fact that ladders with equal termination resistances are least sensitive to individual element tolerances. Again, in (15), $\omega_N$ is a normalization frequency.

$$\omega^2 = \frac{\omega^2 - \omega^2_0}{\omega^2 - \omega^2_0}$$

(15)

Finally, equation (16) is the combination of (14) and (15), with $k$ being the frequency normalization constant. This combination performs both transformations simultaneously.

$$\omega^2 = \frac{k(\omega^2 - \omega^2_0)}{\omega^2 - \omega^2_{max}} \bigg| \frac{\omega^2 - \omega^2_0}{\omega^2 - \omega^2_0} \bigg|$$

(16)

Figure 3 presents the transfer function and some relevant data, as generated by LADDER. Figure 4 shows the magnitude response, transient response, and s-plane, with the unit circle and pole/zero locations.

Synthesis of a ladder network follows the procedure given herein. Because this example has three conjugate zero pairs, six circuits, as shown in Table 1, are possible. LADDER draws a circuit diagram of the synthesized circuit, Figure 5 for Set 1.
LOWPASS - Cauer elliptic 7 order

POLYNOMIAL:

\[ \text{Table 1. 6 circuits for a } 7\text{th order lowpass Cauer filter with } R_{out} = 2 \]

<table>
<thead>
<tr>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inductor</td>
<td>Capacitor</td>
<td>Inductor</td>
</tr>
<tr>
<td>0.000</td>
<td>3.274</td>
<td>0.000</td>
</tr>
<tr>
<td>0.650</td>
<td>0.368</td>
<td>0.650</td>
</tr>
<tr>
<td>0.000</td>
<td>4.242</td>
<td>0.000</td>
</tr>
<tr>
<td>0.721</td>
<td>0.224</td>
<td>0.769</td>
</tr>
<tr>
<td>0.000</td>
<td>4.509</td>
<td>0.000</td>
</tr>
<tr>
<td>0.728</td>
<td>0.072</td>
<td>0.652</td>
</tr>
<tr>
<td>0.000</td>
<td>3.799</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Monte Carlo analysis completes the synthesis procedure. Built-in Monte Carlo analysis routines give a quick visual demonstration of a synthesized filter's sensitivity. These routines use known characteristics of ladder forms for a more streamlined approach than general filter analysis programs. The quick, built-in analysis routines are especially important for designs with multiple solutions. In the case of filters with multiple solutions, some solutions will not give practical, well-scaled element values; however, when several solutions offer a practical circuit, ladder sensitivity is the next logical criteria. Figure 6 illustrates the Monte Carlo simulation for Set 1, as produced by LADDER.

CONCLUSION

An algorithm for the synthesis of analog ladder filter prototypes was developed. The proposed approach is general. In the case of Butterworth, Chebyshev, and Bessel-Thompson filters, the complex continued fraction algorithm reduces to a classical continued fraction. The generality of this approach also allows the synthesis of transfer functions generated using non-traditional methods.

REFERENCES

Proceedings of the
35th Midwest Symposium
on Circuits and Systems

August 9 - 12, 1992
Washington, D.C.