SAFIMAM: Synthesis of Analog Filters in MATLAB by Approximation Methods

1. Software Requirements

This software only requires a basic Matlab setup with Symbolic Math Toolbox included. To design filters with classical approximation methods, the software requires the toolbox which is located in the Functions_of_classical_methods/AF_TOOLBOX_2016 folder. The toolbox supporting the book: Lars Wanhammar; Analog Filters Using MATLAB, Springer, 2009.

2. Installation

The software does not need any installation. It is sufficient to add the folders with subfolders to the Matlab path.

3. Folder Organization

The software is organized in several folders. The src folder includes fig and m files for the SAFIMAM interface, in these files are contained all the interface windows that have the functions of determining the low-pass filter order, the low-pass filter transfer functions, the low-pass frequency responses, the low-pass electrical networks, and the low-pass filter transformations to other response types. The Pascal_Functions folder contains the functions used to determine the order of the Pascal filters, the transfer functions for the responses with optimization in the pass band and optimization in the stop band, and also includes the function that performs the synthesis to determine the values of the passive electrical networks of the filters, as well as the functions that complement it. The folder Functions_of_classical_metods contains the toolbox AF_TOOLBOX_2016 supporting the book Lars Wanhammar: Analog Filters Using MATLAB, Springer, 2009, which contains the functions to perform the synthesis of the transfer functions of the classical approximation methods. Finally, in the Electrical_Network folder are stored the images of the different passive electrical networks of the filters, low pass, high pass, band pass and band reject, in the T-type and π-type network configurations, these networks are the ones projected in the interface when determining the electrical networks of the filters, with the objective of visualizing the generated electrical network.

4. Start Guide

This section briefly explains the operation of the SAFIMAM interface.

Once the folders contained in the repository have been added to the Matlab path, simply execute the word “order” in the Matlab command window to open the SAFIMAM interface start window.

4.1 Calculating filter order

Once the first window of the interface is opened, the window corresponds to the section in which the filter orders will be determined as shown in Fig. 1.

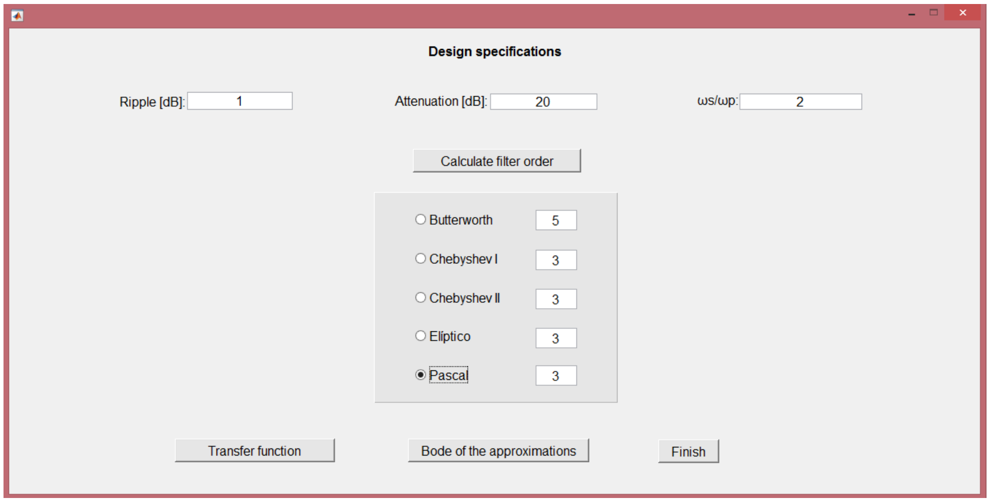

Figure 1. Menu design specifications.

As can be seen in the window of Fig. 1, three design specifications must be entered, which are ripple, attenuation and transition ratio, in order to subsequently determine the order of the filters with each approximation method. At the bottom of the window there are buttons to calculate the transfer function of the selected filter or to determine the Bode response of all approximation methods.

4.2 Calculating the frequency response of all filters

If the user chooses to observe the responses of all approximation methods for comparison purposes, the window in Fig. 2 opens.

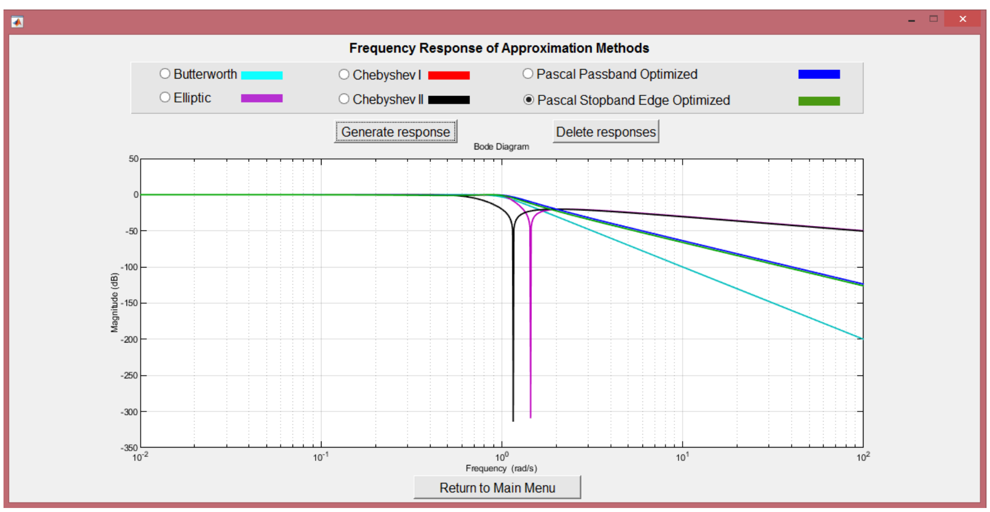

Figure 2. Frequency response of all approximation methods.

In the window in Fig. 2, the user can compare the methods he wants.

4.3 Filter design with Pascal approximation method

If the user chooses the Pascal approximation in the Design Specifications menu window and wants to calculate the transfer function, it opens up the advantage of Fig. 3, where the calculation can be performed with the ideal case and the real case.

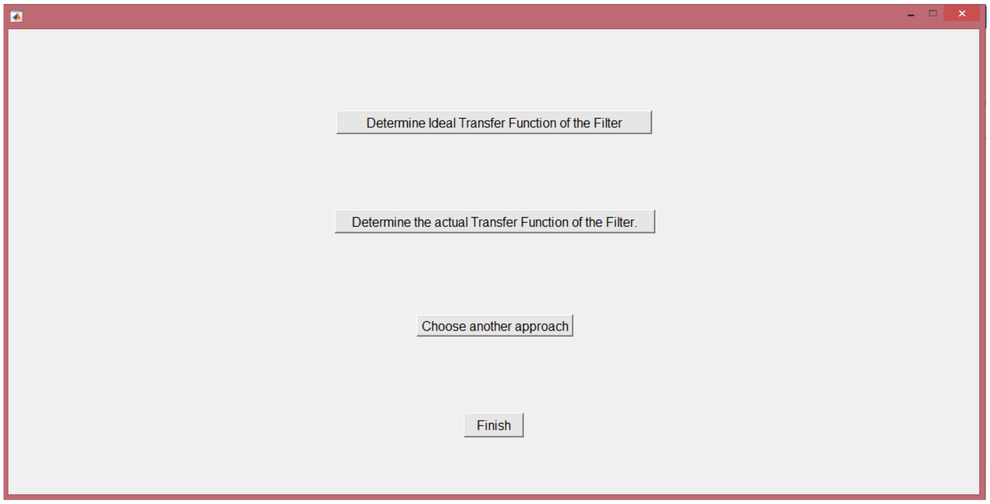

Figure 3. Pascal filter design, real and ideal case.

4.3.1 Pascal Transfer Function ideal case

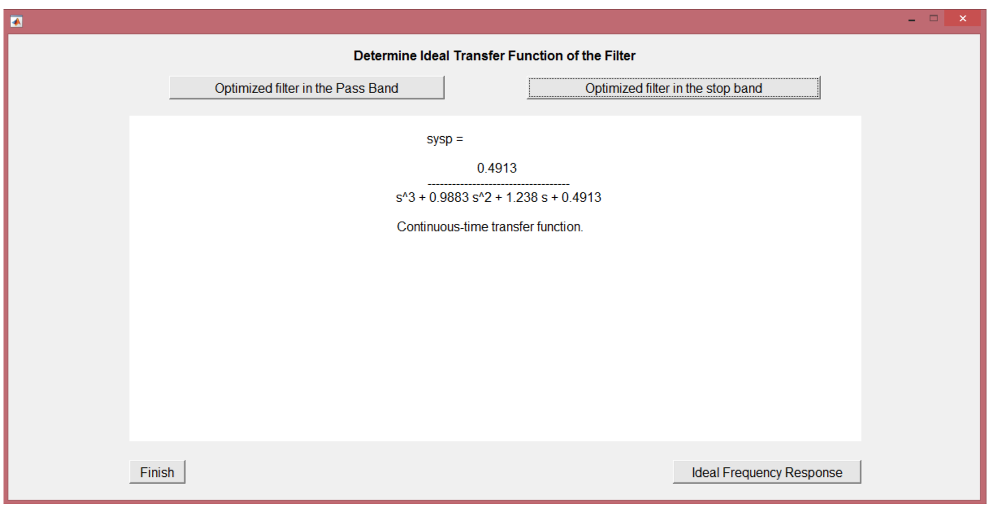

Figure 4. Pascal's ideal transfer functions.

When the ideal case is chosen, the transfer function can be calculated for the Pascal variation with optimization in the pass band and optimization in the stop band as shown in Fig. 4, and finally the frequency responses of the cases can be generated as shown in Fig. 5 and return to the menu of Fig. 3 to design the actual filter.

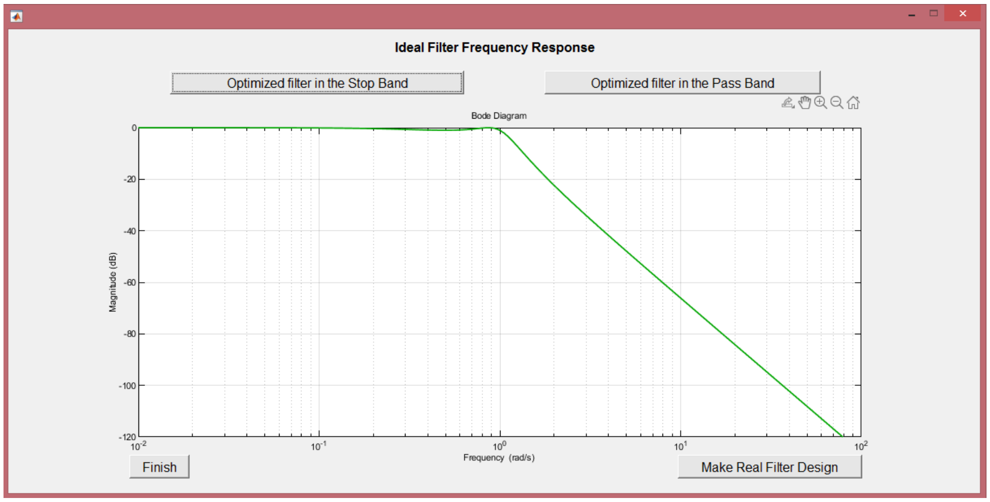

Figure 5. Pascal's ideal frequency responses.

4.3.2 Calculating transfer function of Pascal low-pass filter

Choosing the actual filter design from the menu in Fig. 3 opens the window in Fig. 6 for determining the filter transfer function, where the specification of the internal resistance Rs of the power grid must be entered that will be obtained at the end of the design.

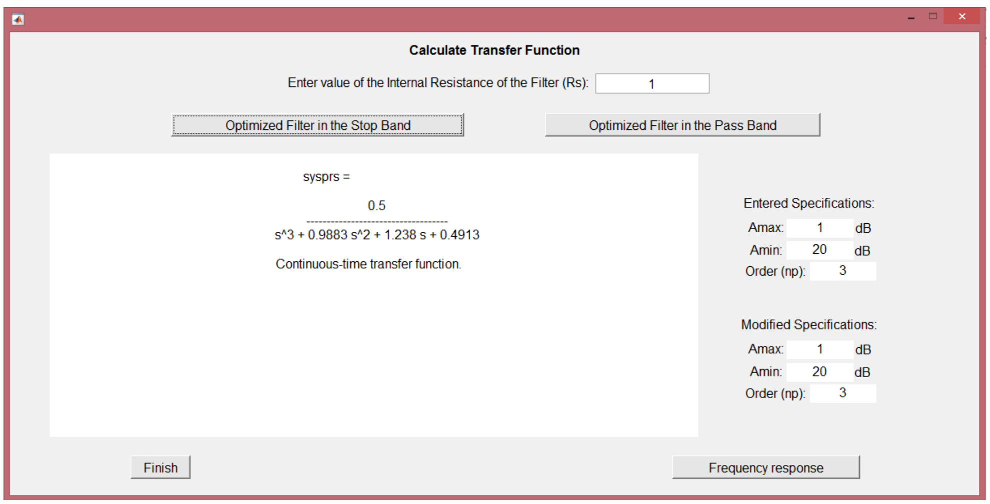

Figure 6. Low Pass Pascal transfer function calculation.

Fig. 6 shows a section of specifications entered by the user and modified specifications, this is because in the design of even order filters, the specifications can be automatically modified in case the filter cannot be designed directly with the specifications indicated at the beginning, the software is able to modify these specifications in order to obtain an even order filter and as a last resort it can modify the order to the next odd order if the filter cannot be designed directly.

4.3.3 Calculating frequency response of the Pascal low-pass filter

Once the transfer functions have been calculated, the frequency responses of the two Pascal variations can be calculated as shown in Fig. 7.

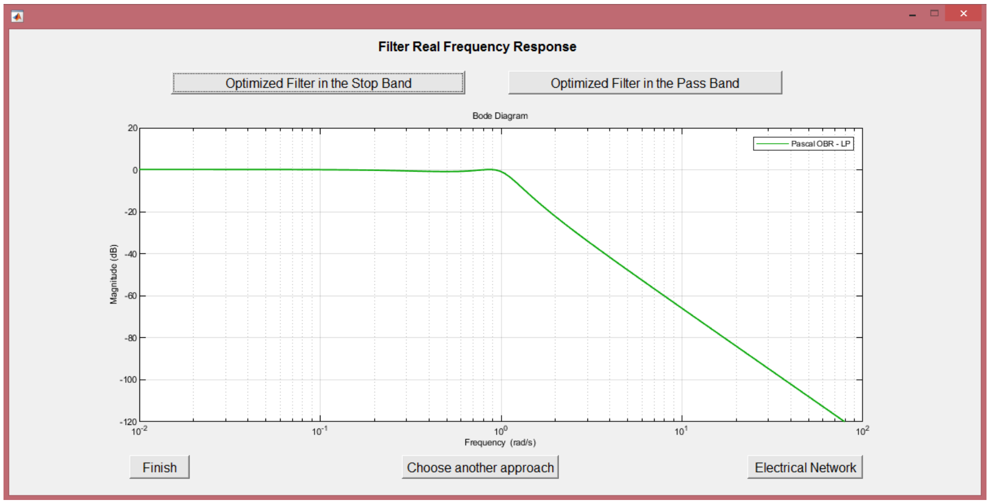

Figure 7. Low pass frequency response of Pascal filters.

4.3.4 Calculating passive low-pass electrical networks Pascal

Subsequently, the low-pass electrical networks of the Pascal variations can be calculated as shown in Fig. 8, both in the T-type network configuration and the π-type network. On the one hand, the image of the electrical network configuration is shown and on the other side the values of the resistors, inductors and capacitors that make up the network are shown.

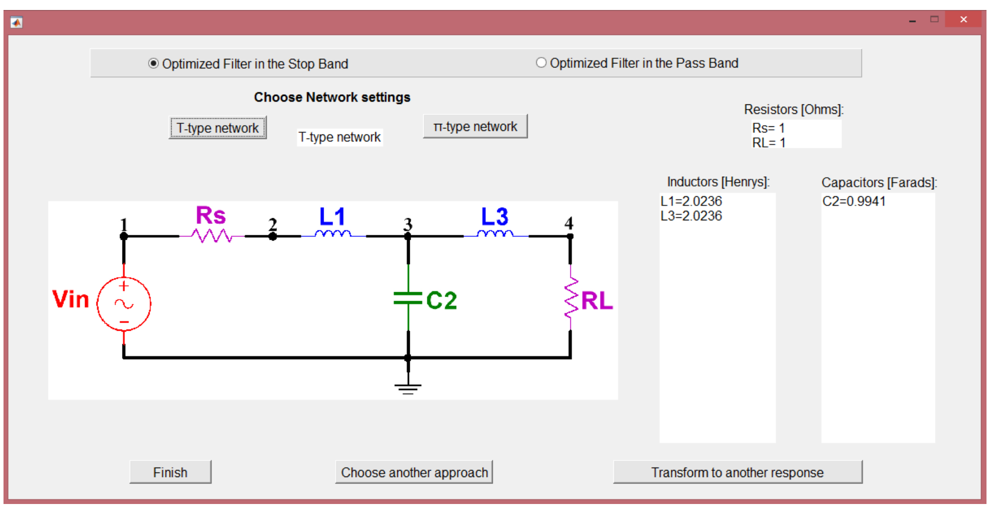

Figure 8. Electrical Networks of Pascal low pass filters.

4.3.5 Pascal Low Pass Filter Transformation Options

Once the design of the Pascal low pass filter is completed, if the user wishes, they can transform this filter to other types of responses. Therefore, choosing this option opens the window in Fig. 9, where the transformation options menu is displayed.

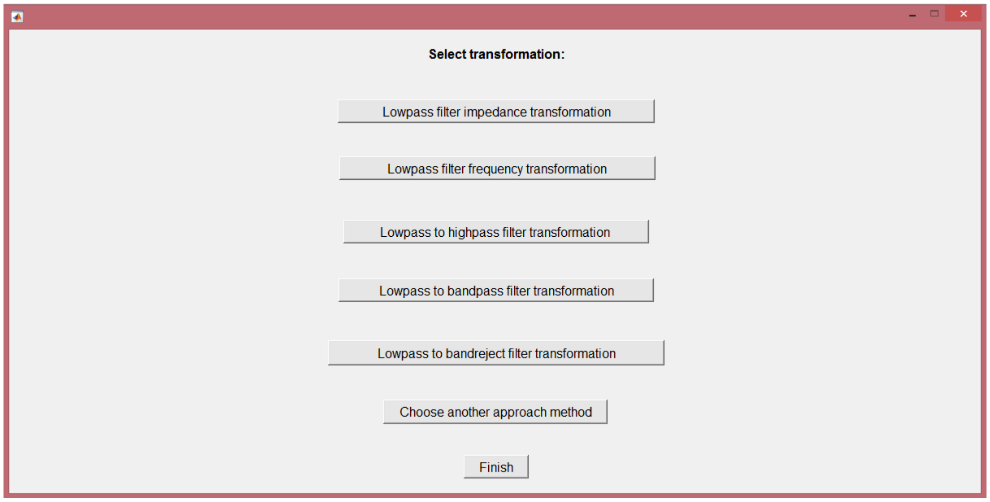

Figure 9. Pascal low pass filter transformations menu.

4.3.6 Transformation into low-pass filter impedance Pascal

When choosing the impedance transformation from the menu in Fig. 9, the window in Fig. 10 opens, where the required denormalization impedance must be entered, to subsequently calculate the transfer function.

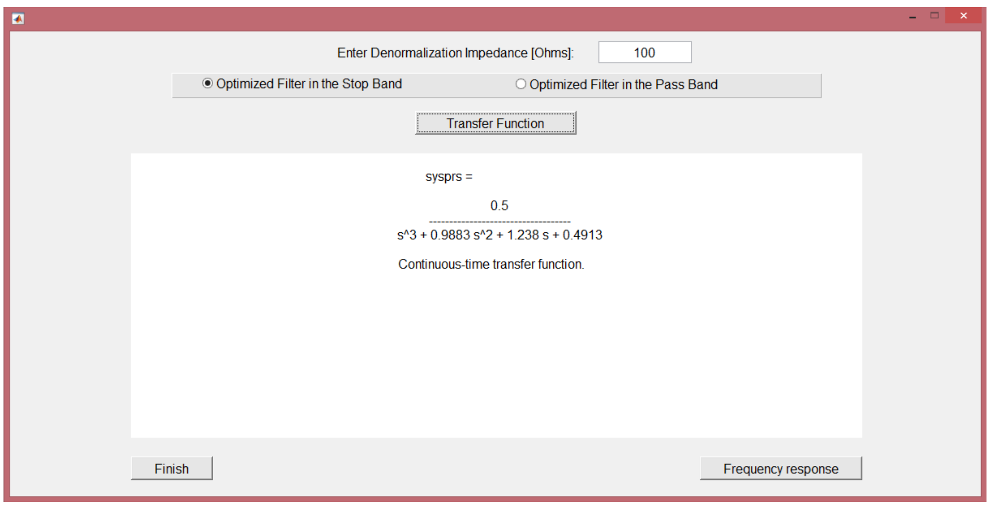

Figure 10. Transformation into impedance of the Pascal low-pass filter.

Once the transfer function is calculated, the frequency response can be generated as shown in Fig. 11 and the electrical network as shown in Fig. 12.

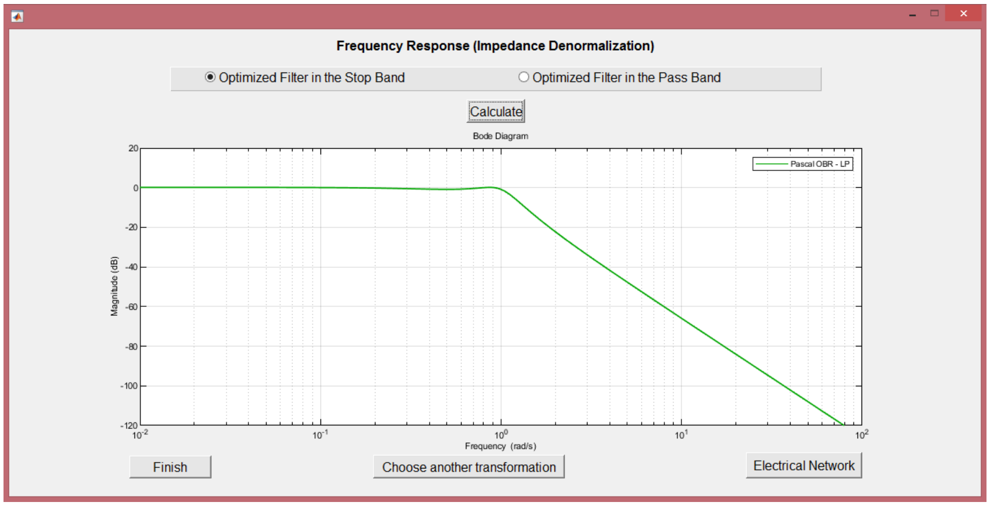

Figure 11. Frequency response of the impedance transformation of the pascal low-pass filter.

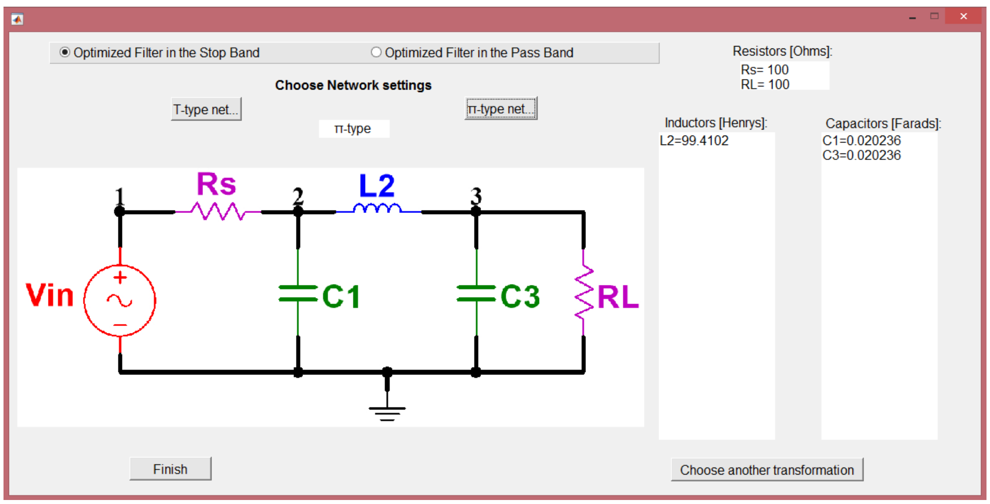

Figure 12. Electrical networks low pass pascals denormalized in impedance.

4.3.7 Pascal low-pass filter frequency transformation

When choosing the frequency transformation from the menu in Fig. 9, the window in Fig. 13 opens, where the required denormalization frequency must be entered, to subsequently calculate the transfer function.

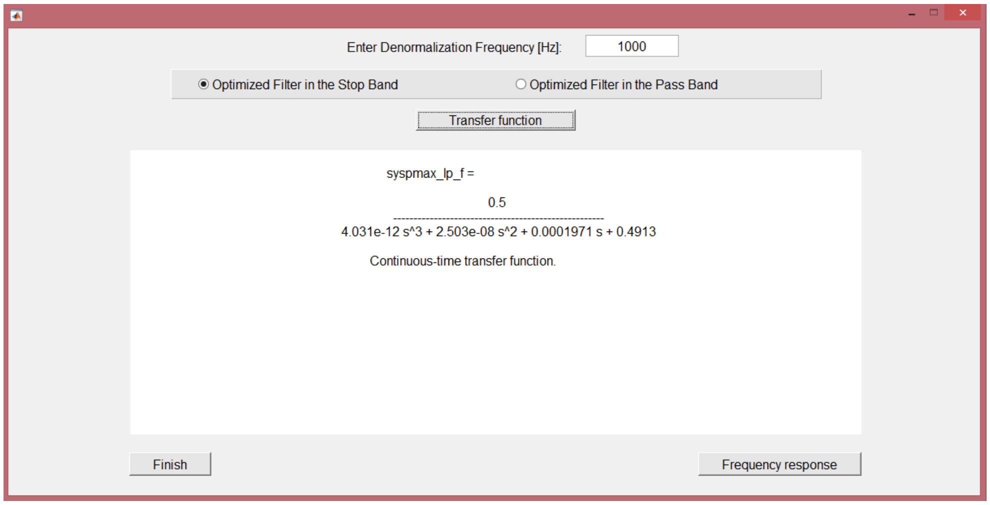

Figure 13. Transformation into impedance of the Pascal low-pass filter.

Once the transfer function has been calculated, the frequency response can be calculated as shown in Fig. 14, and the filter's electrical network transformed into frequency as shown in Fig. 15.

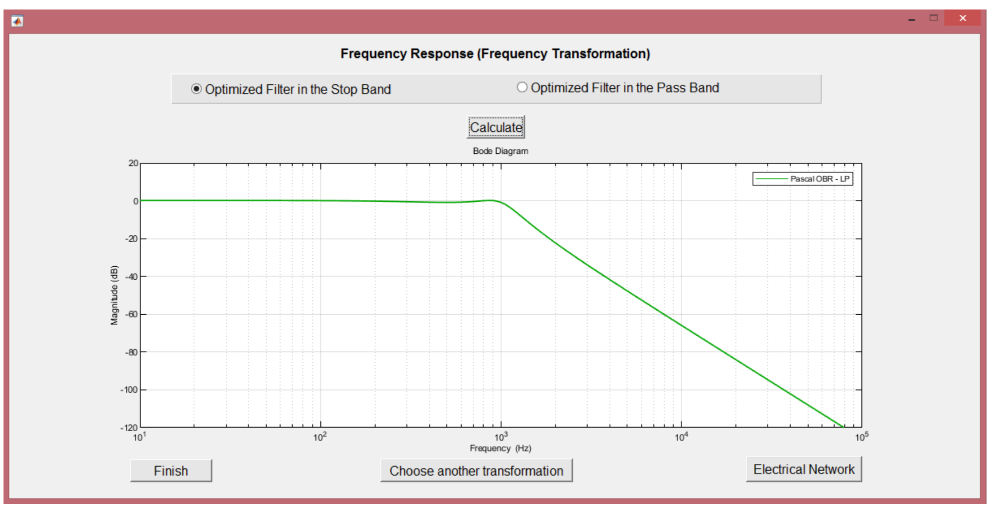

Figure 14. Frequency responses of the low-pass Pascal filter transformed into frequency.

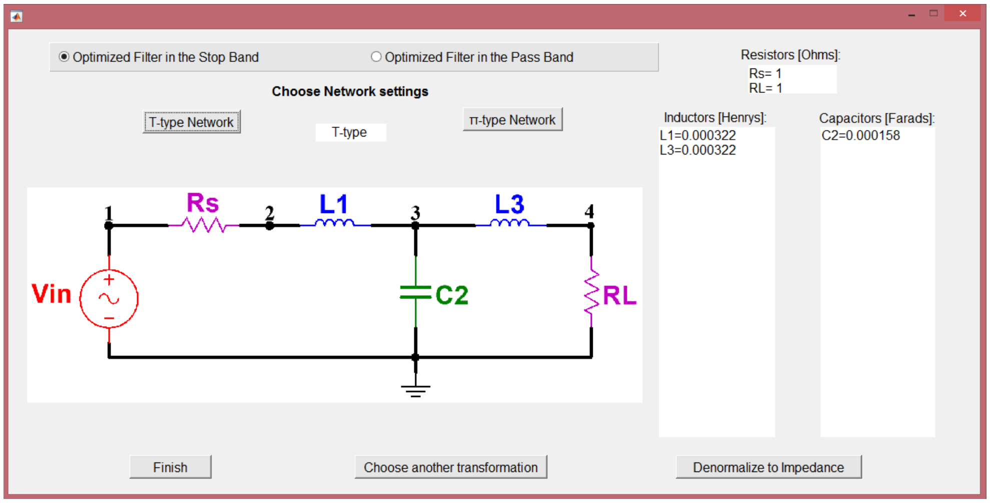

Figure 15. Electrical Networks of the low-pass Pascal filter transformed into frequency.

Once the frequency transformation of the filter has been performed, the user can continue with the design by performing the impedance transformation. If the user chooses to perform the transformation, the advantage of Fig. 16 opens, where the denormalization impedance must be entered to determine the transfer function, the frequency response (Fig. 17) and finally the resulting electrical network (Fig. 18). When obtaining the impedance denormalized electrical network from the frequency transformation, the user can return to the transformations menu of Fig. 9 to be able to perform any other transformation desired.

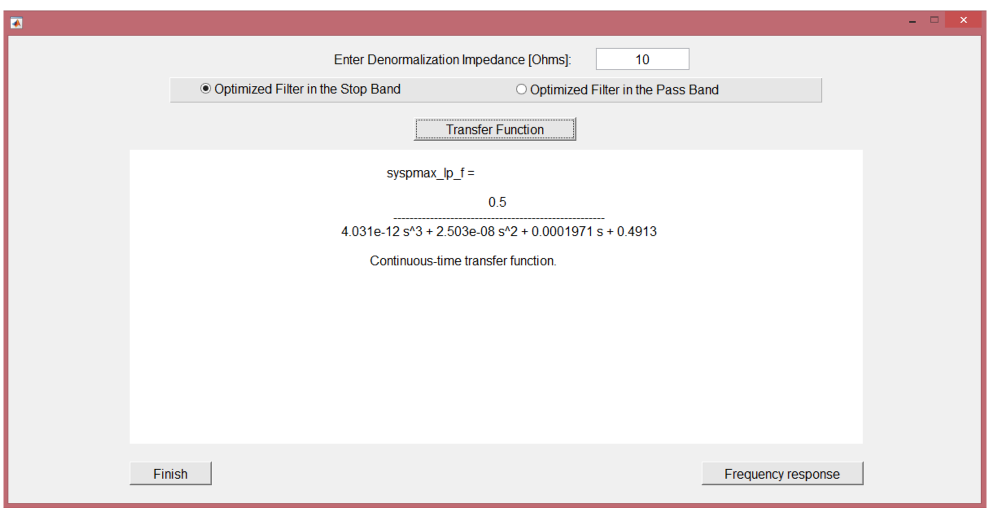

Figure 16. Impedance denormalization of the Pascal filter transformed into frequency.

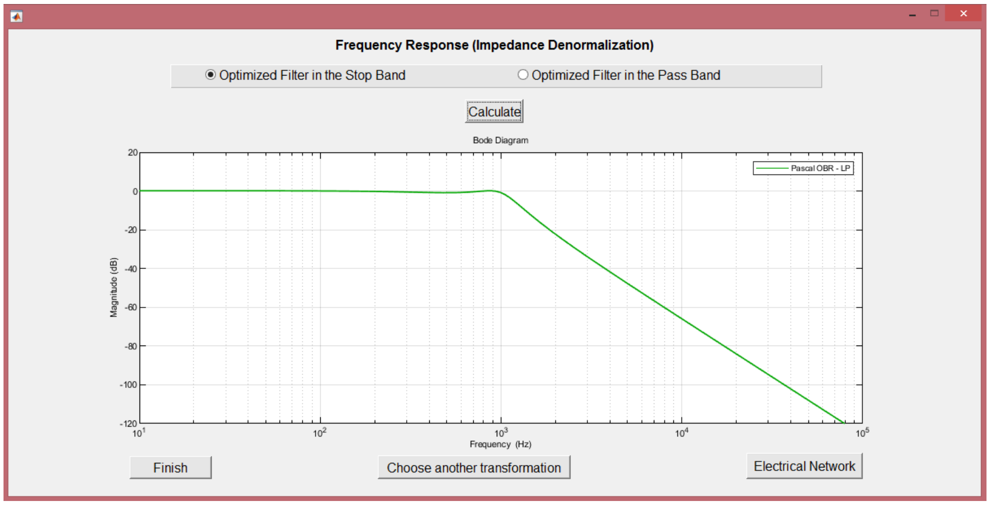

Figure 17. Frequency response of the impedance denormalization of the frequency transformed Pascal filter.

4.3.8 Low-pass to high-pass Pascal filter transformation

When choosing the transformation to high-pass filter from the menu in Fig. 9, the window in Fig. 19 opens, where the required cut-off frequency must be entered, to later calculate the transfer function.

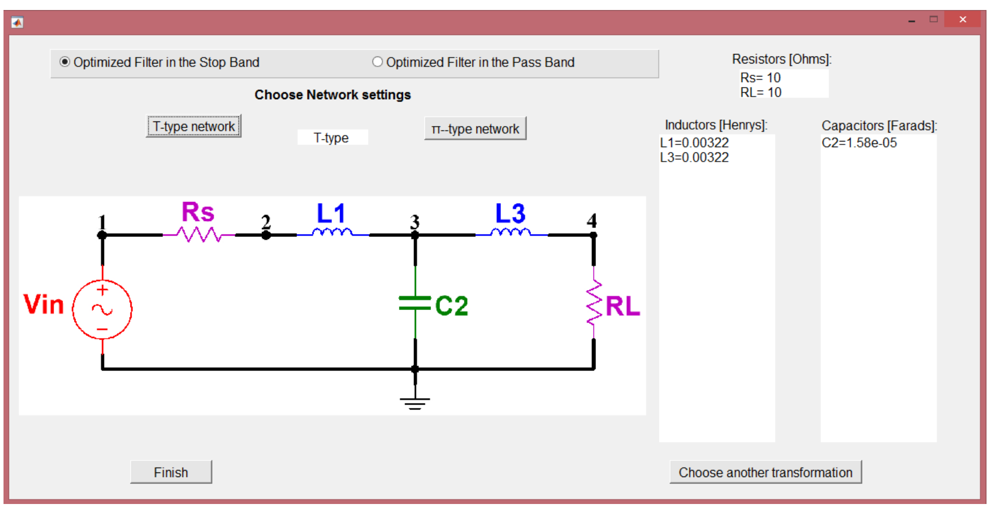

Figure 18. Electrical Networks denormalized in impedance of the Pascal filter transformed into frequency.

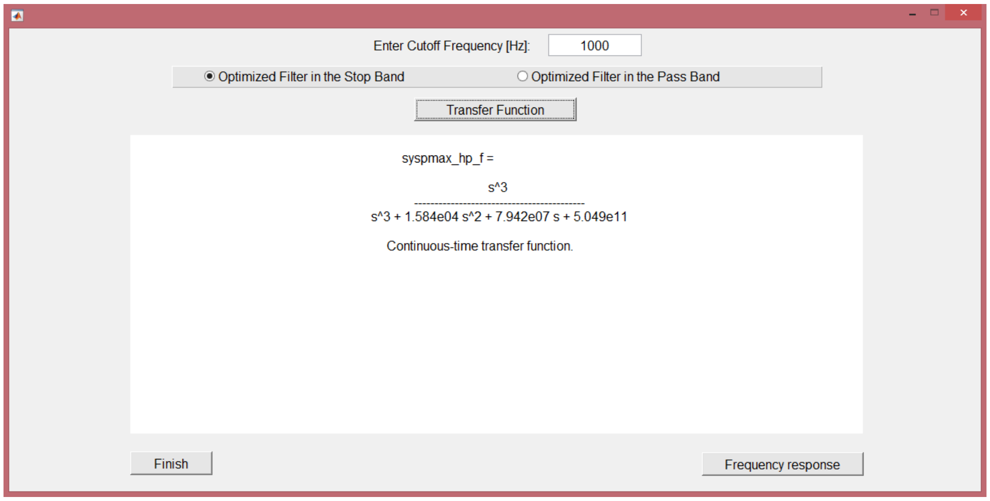

Figure 19. Transformation of the Pascal filter from low pass to high pass.

Once the high-pass transfer function has been calculated, the frequency response can be calculated as shown in Fig. 20 and the electrical networks of the filters as shown in Fig. 21. Subsequently, the impedance denormalization of the high-pass filter can be performed by entering the denormalization impedance as shown in the window of Fig. 22, and the frequency response and electrical network can be calculated as shown in Fig. 23 and Fig. 24, respectively.

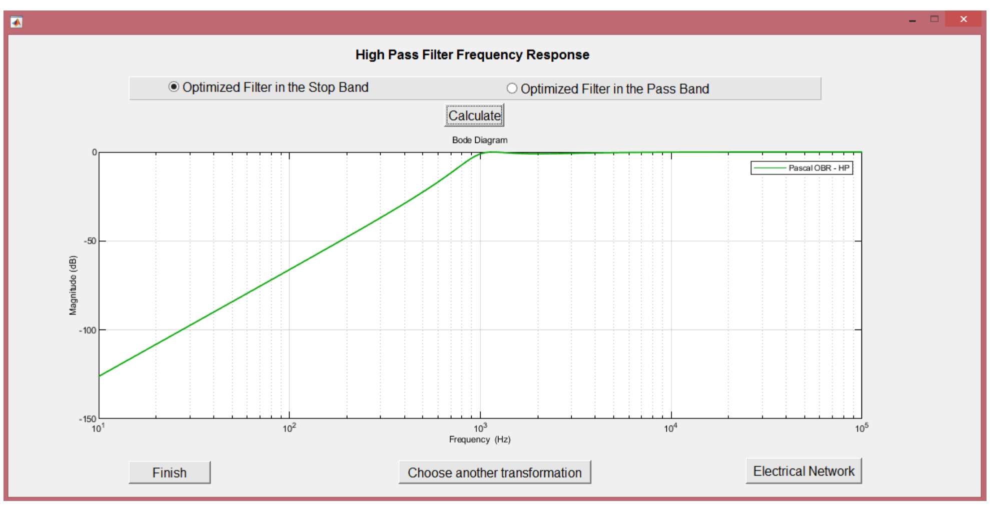

Figure 20. Frequency response of the high-pass Pascal filter.

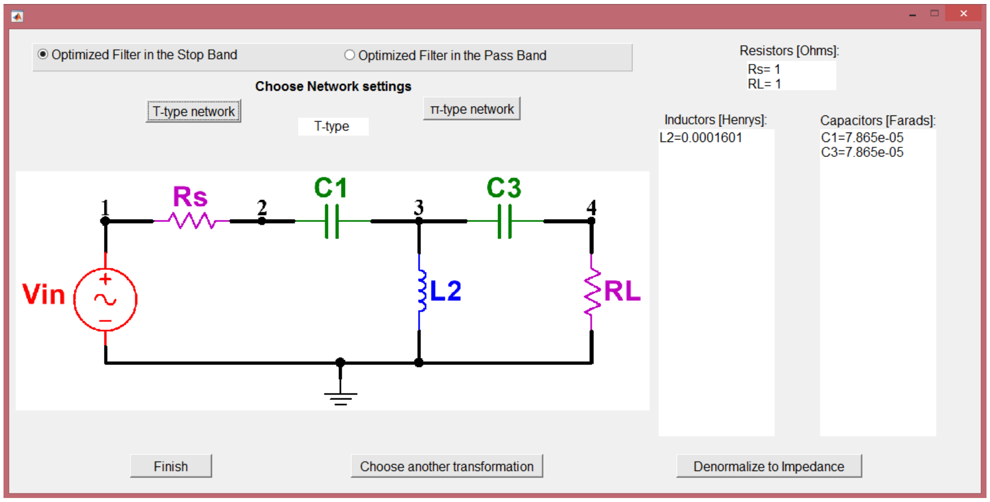

Figure 21. Electrical networks high pass Pascal filter.

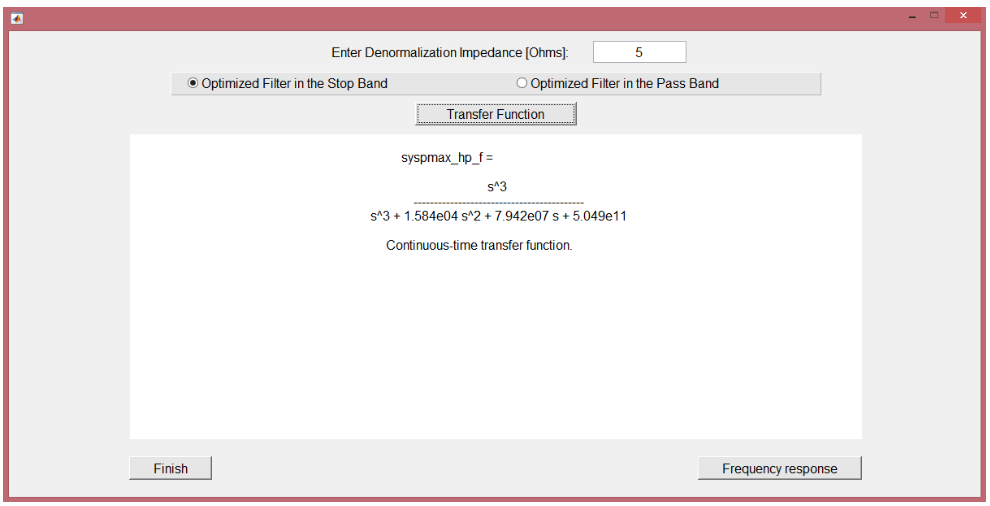

Figure 22. Impedance denormalization of the high-pass Pascal filter.

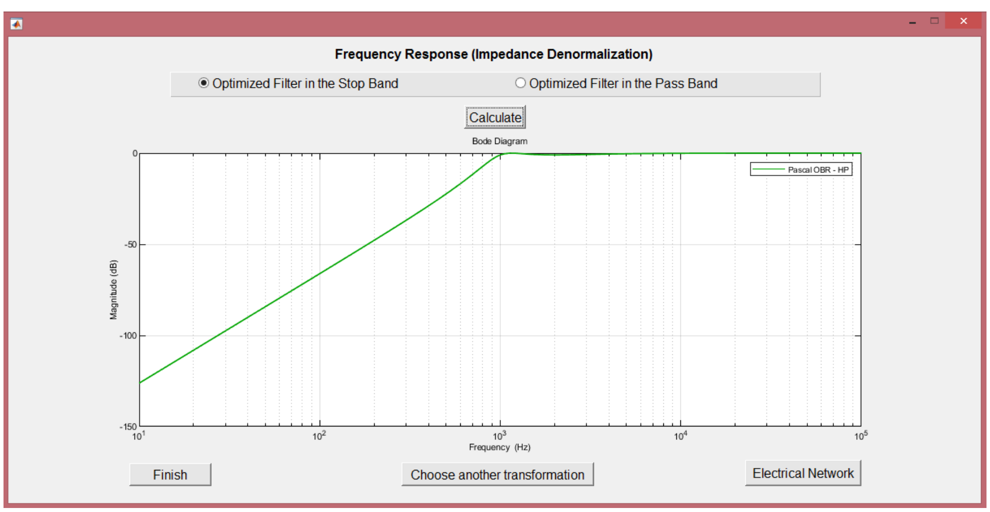

Figure 23. Frequency response of the impedance-denormalized high-pass filter.

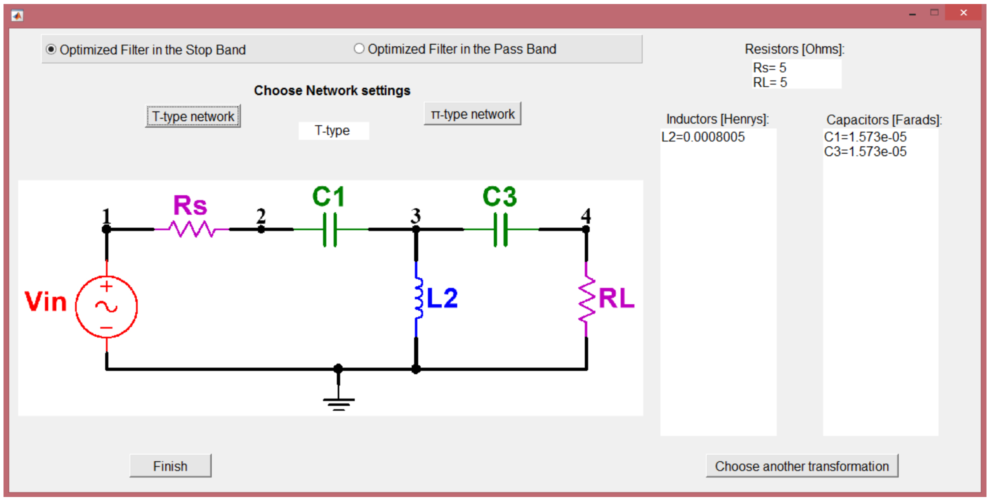

Figure 24. Electrical networks of the high-pass Pascal filter denormalized in impedance.

4.3.9 Low-pass to band-pass Pascal filter transformation

When choosing the transformation to band-pass filter from the menu in Fig. 9, the window in Fig. 25 opens, where the required cut-off frequencies must be entered, to subsequently calculate the band-pass type transfer function.

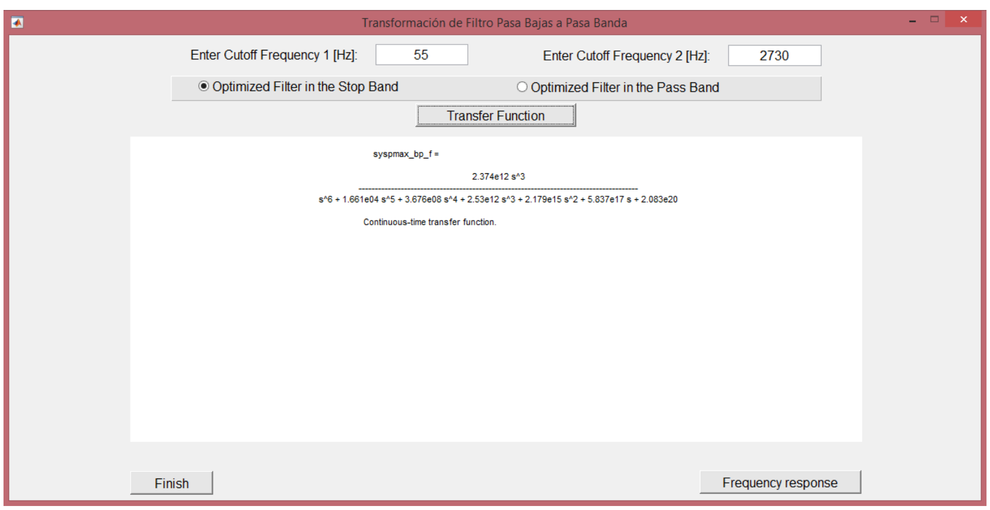

Figure 25. Low-pass to band-pass Pascal filter transformation.

Once the band-pass transfer function has been calculated, the frequency response can be calculated as shown in Fig. 26 and the electrical networks of the filters as shown in Fig. 27.

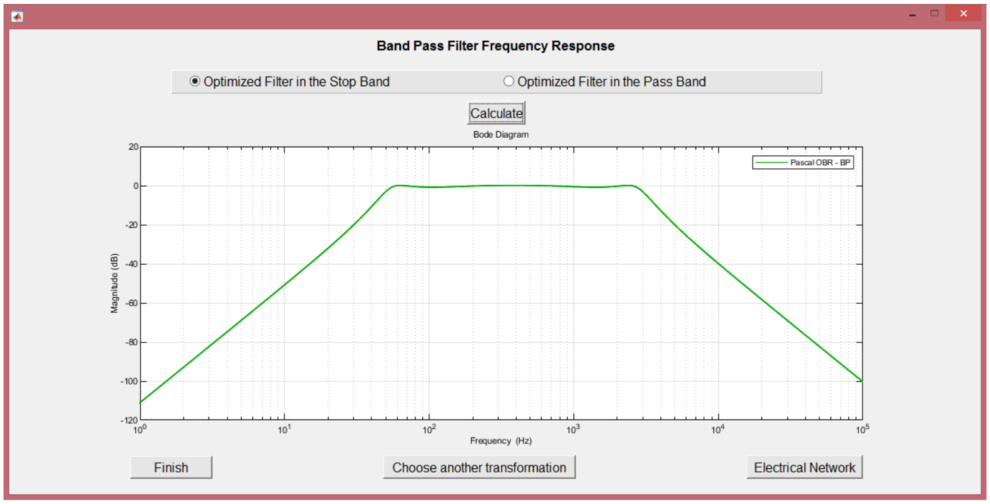

Figure 26. Frequency response of the Pascal band pass filter.

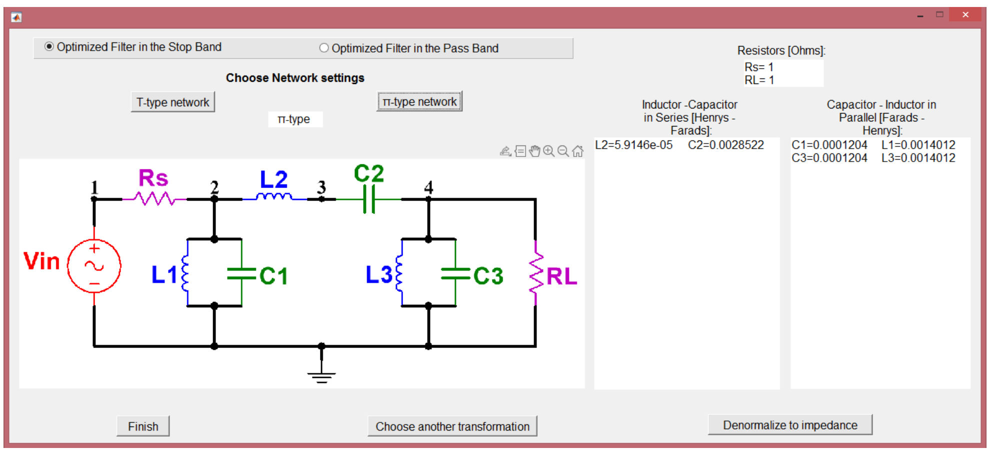

Figure 27. Electrical networks of the Pascal band pass filter.

Subsequently, the impedance denormalization of the band pass filter can be carried out by entering the denormalization impedance, in the same way calculating the frequency response and the electrical network, as in the case of the transformation of the low pass filter to high pass.

4.3.10 Low-pass to band-reject Pascal filter transformation

When choosing the band reject filter transformation from the menu of Fig. 9, the window shown in Fig. 28 opens, where the required cut-off frequencies must be entered, in order to subsequently calculate the band reject type transfer function. Once the band reject type transfer function has been calculated, the frequency response can be calculated as shown in Fig. 29 and the electrical networks of the filters as shown in Fig. 30.

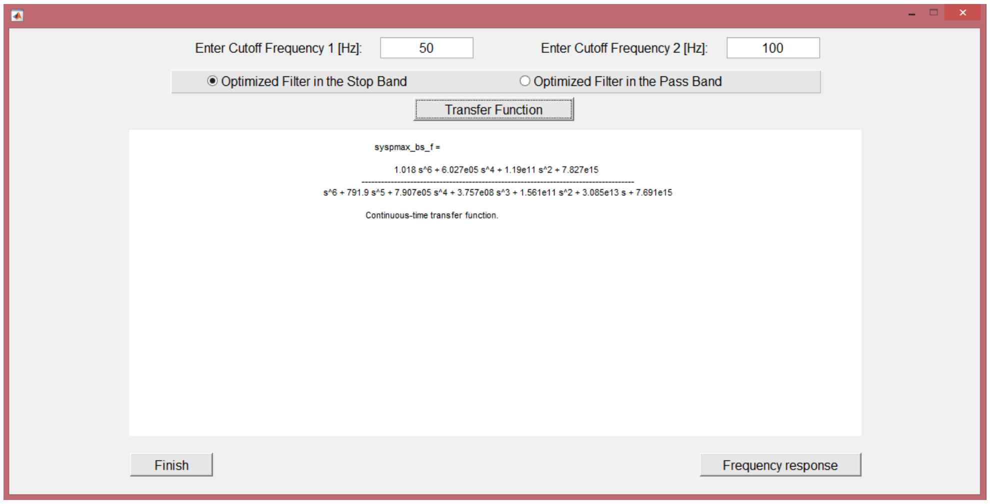

Figure 28. Low-pass to band-reject Pascal filter transformation.

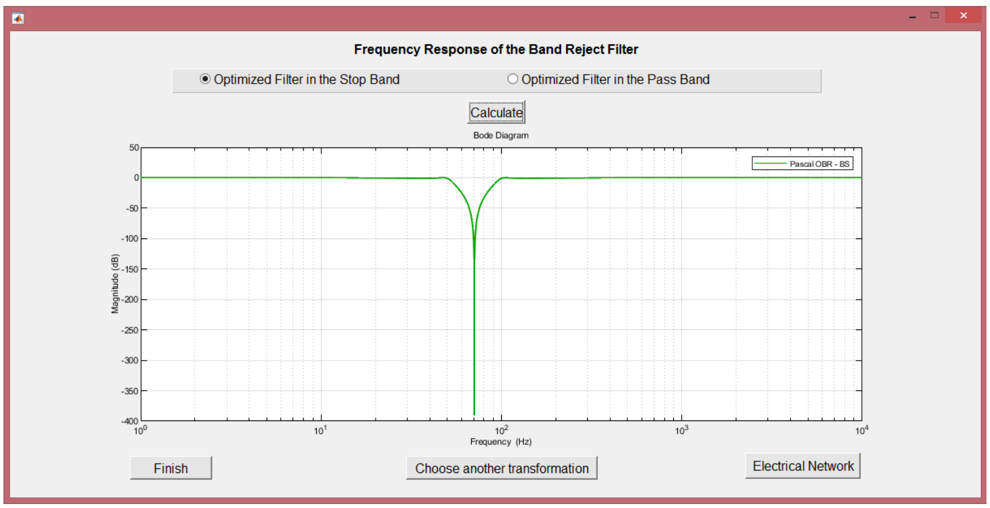

Figure 29. Frequency response of the Pascal band reject filter.

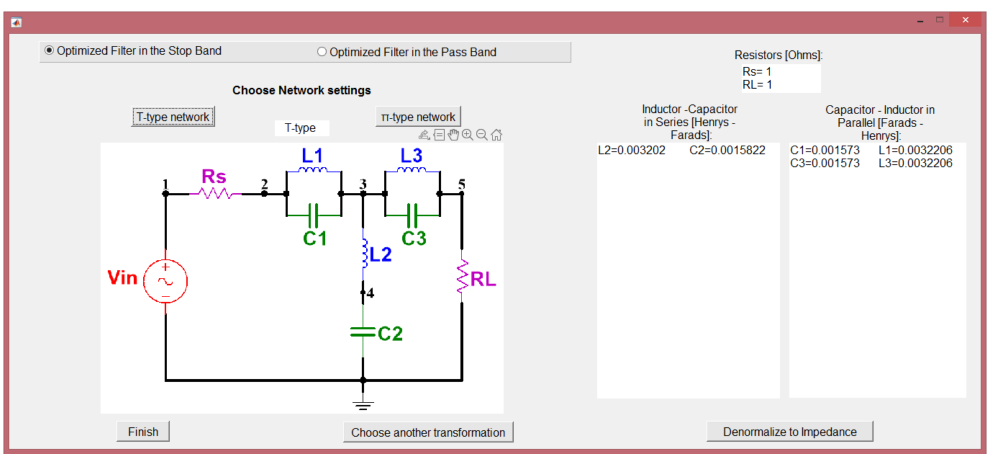

Figure 30. Electrical networks of the Pascal band reject filter.

Subsequently, the impedance denormalization of the band reject filter can be carried out by entering the denormalization impedance, in the same way calculating the frequency response and the electrical network, as in the case of the transformation of the low-pass filter to high-pass.

4.4 Filter design with classical approximation methods

For the design of filters with classical approximation methods, the dynamics is similar to the design of Pascal filters explained above, in the menu of Fig. 1 when the user chooses one of the classical approximation methods to perform the design of a filter, for example, the case of a Butterworth filter as shown in Fig. 31.

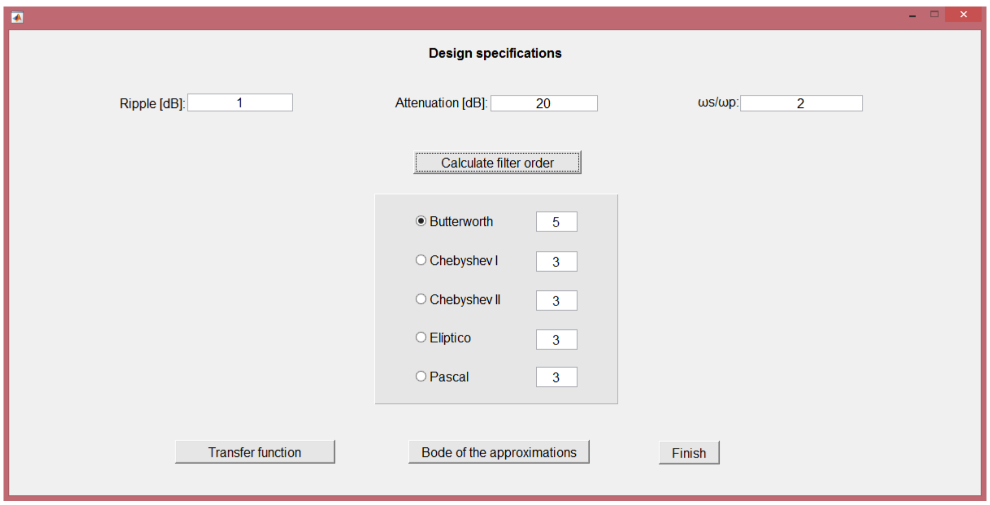

Figure 31. Filter design with the Butterworth approximation method.

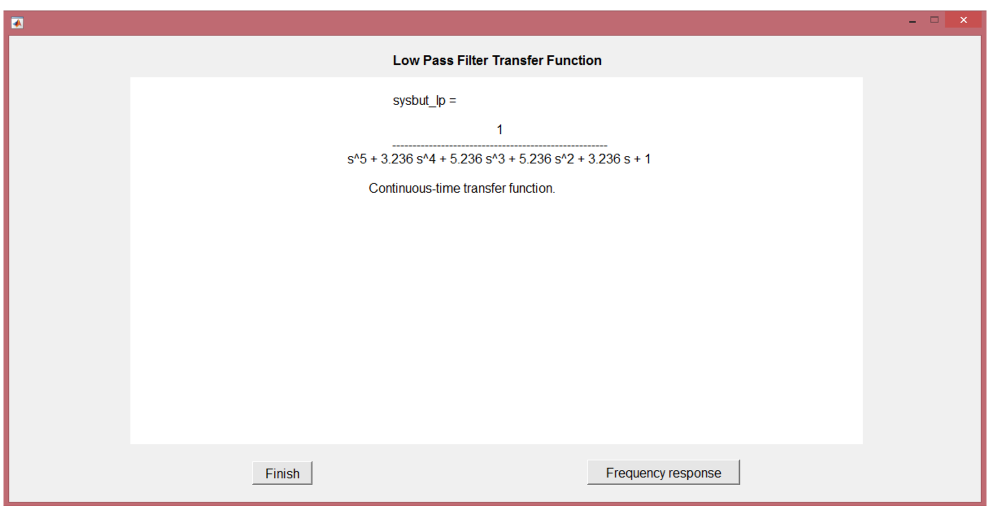

Figure 32. Butterworth low pass filter transfer function.

By choosing the desired method, the user can calculate the transfer function as shown in Fig. 32, then generate the frequency response as shown in the window of Fig. 33 and finally the electrical network of the low pass filter with passive elements as shown in Fig. 34, where the value of the internal resistance Rs of the passive electrical network to be obtained must be entered and saved.

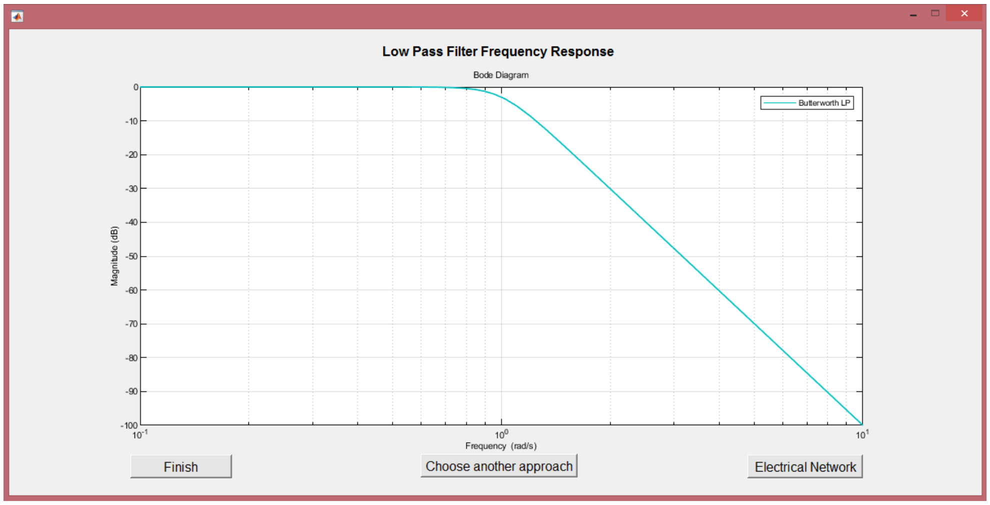

Figure 33. Frequency response of the Butterworth low pass filter.

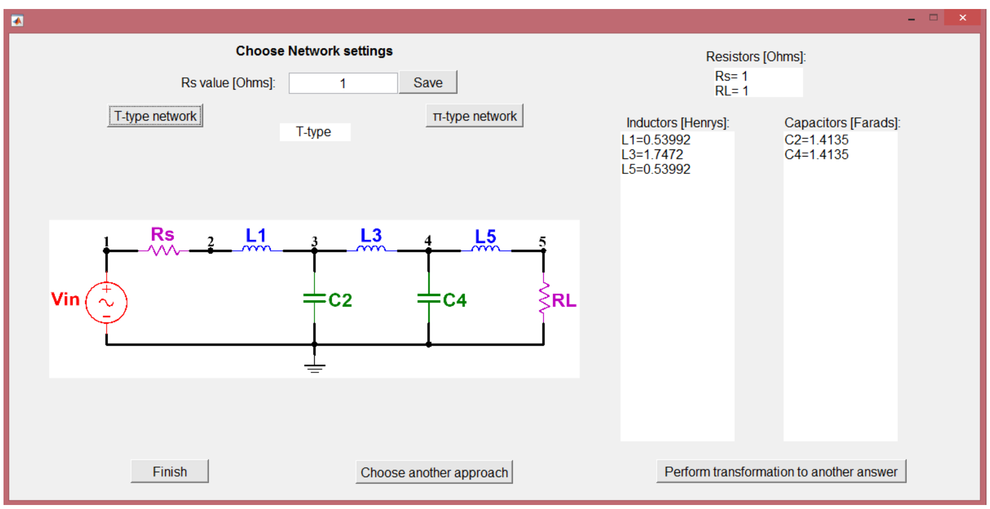

Figure 34. Butterworth low pass filter electrical networks.

Once the low-pass filter design is finished with the classical methods, if the user wishes, he can transform this filter to other types of responses, so when choosing this option a window opens with the menu of available transformations as shown in Fig. 9, when choosing one of them the design process is similar to that shown in the Pascal case for all available transformations, either a transformation in frequency, impedance, or even to other types of responses such as high-pass, band-pass and band-reject.