Design and Implementation of Optimized Filters

João Pedro Gomes Moura Azevedo
j.pedro.azevedo@tecnico.ulisboa.pt
Instituto Superior Técnico, Lisboa, Portugal
June 2023

***Abstract*—Analog filters are present in several applications such as, for example, noise removal in audio processing, signal spectrum limitation in A/D converters or biological signals conditioning. A filter is designed to meet specifications in the frequency and/or time domain. The objective of this work is to design and implement analog filters in an optimized and automated way. A tool for the design and implementation of passive and active filters, with characteristics specified by an user, was developed. The tool produces a PDF report with the description of the solution produced (transfer function computation, circuit selection and component sizing). The existence of several optimization criteria requires a compromise between them, making the project a multi-objective problem. Optimization is necessary to improve the cost-benefit ratio of the obtained solution, from the choice of the approximation to minimize the filter order, to the comparison between the different designs to minimize the sensitivity of the filter parameters. In implementations with available standard components, it is also necessary to consider the deviation in the value of the components parameters and their respective associated tolerance, which allows a statistical analysis to be carried out to understand the expected variation of the response in the mass production of filters. The developed tool is illustrated with an example of an active bandpass filter project and the result is compared with one of the tools available from a manufacturer, allowing to illustrate the reduction in the deviation margin of the gain and the group-delay.**

***Index Terms*—analog filter, filter project tool, multi-objective problem, filter optimization/automation, sensitivity analysis, standard components**

I. Introduction

In electrical and electronic engineering, an analog filter may be generically considered as a circuit designed to pass a desired band of frequencies, while rejecting all signals outside that band of interest [1] [2] [3]. However, there are other objectives that can be considered in an application with filters, such as the characteristics of the time response to a certain input signal, or a group-delay response with certain characteristics in the frequency range of interest [1] [4]. Most electric and electronic circuits, nowadays, vastly use analog filters in many modern applications, such as noise removal in audio processing, anti-aliasing in A/D converters, signal enhancement in biomedical applications, etc. [1] [5]. Although a lot of systems have moved into the digital domain in the last decades, the world is still fundamentally analog in nature, and the need of signal conditioning through a filtering function remains present [1].

There are multiple criteria that might need to be achieved by a filter, depending on its application and specifications. The adjustment of filter parameters is rarely a single-objective task, because the different components affect multiple parameters, as well as the various decisions taken in the project flow regarding the transfer function or the network used to implement the circuit. The most common design process is to find a transfer function that meets the most important requirements and specifications (which, of course, depend on the application), and then to decide on an appropriate topology that implements it [1]. While during most of the design process all component values are assumed to exist in an ideal continuous interval, according to the IEC 60063:2015 norm, the manufactured passive components (resistors, inductors and capacitors) follow a standard list of defined values, approximately logarithmic, and its multiples, which are collected in what are known as the "E series" [6]. A higher number of values within a series implies more precision, but an increase of cost [6]. Now, while some components can be chosen to be indeed equal to the standard values available (or a series/parallel association of those components), inevitably some rounding will be needed for the others, which will introduce some kind of error in the desired filter response [6]. Furthermore, when active filters are being considered, there are additional limitations and non-ideal effects of the operational amplifier (OpAmp), that may introduce errors (the finite gain-bandwidth product (GBW), for instance), which also need to be addressed in the choice of the model of the OpAmp [1].

Filter optimization is a multi-objective optimization (MOO) problem, because it is defined as "the simultaneous optimization of several fitness functions that are often in conflict with each other"(citing from [7]).

The improvement of optimization and automation on the design and implementation of analog filters is, therefore, an important field of study with practical application. For this reason, manufacturers, along the years, have developed some software tools in order to make this process easier and quicker for engineers that are less aware about filter design and implementation methodologies.

II. Objectives

The work developed establishes the foundation, by defining a step-by-step process, to optimize and automate the design and implementation of analog filters with several different structures. Polynomial filters are considered and the choice of the approximation method, passive or active technology, and the different options for the network and components to implement the circuit are considered. The focus of the work is in implementation with standard available components and, therefore, the non-idealities of such components need to be addressed, such as deviations from the nominal value, the tolerances and the frequency-reponse of the active element. In [8] it was already developed a case-study on the optimization of passive filters, which is also here considered and served as the basis for the developed work. The MOO nature of the filter optimization problem is here considered in a broad sense, through the definition of cost functions that tackle different criteria, such as fulfilling certain specifications, or minimizing the sensitivities, leading to a choice of a certain transfer function or a certain network to implement the filter circuit. With all these aspects considered, a tool is developed in *Mathematica* allowing the automation of the process, with the user only needing to provide a list of specifications, and the computation of an optimal solution (transfer function, structure, and list of components and respective tolerances) that implements the desired filter.

In summary, currently, the developed work is a tool that allows optimized and automated design of analog filters, which is independent of the form or support platform where a future application can be implemented, leading to a final product with a graphical interface, such as a filter design and implementation specific software tool or application. Such a tool can be seen as an automatized engineering process that receives specifications and project options as inputs and produces an optimized solution, delivering an output that is a complete circuit for an analog filter implementation.

III. An overview of available filter design tools

Optimizing the design and implementation of filters poses challenges that can make analog filter design a complex process. This motivated manufacturing companies to create software tools that may easily be used by engineers and circuit designers (even though they have little knowledge and little practice in the design and implementation of filters) in order to obtain a filter implementation in a quick simplified manner.

Five software tools currently available for filter synthesis in several technologies were analyzed — the S/FILSYN [9], the LC Simulator from Panasonic [10], the Filter Design Tool from Texas Instruments [11], the FilterLab from Microchip [12], and the Analog Filter Wizard from Analog Devices [13]. They have different characteristics and also different strengths and weaknesses. All provide an easy-to-use interface with the user, as well as different project options. Some are available online (LC Simulator, Filter Design Tool and Analog Filter Wizard), others require installation based on a certain operating system (S/FILSYN and FilterLab). S/FILSYN makes the project assuming ideal filters, that is, it admits the existence of ideal components for the synthesis of a filter, while the other four allow the project with real components, therefore, producing a ready to implement filter. All provide different degrees of optimization. For instance, optimization in the sense that the user can make choices about how to implement the filter, for example, or choosing between a filter with a more precise response (Filter Design Tool and Analog Filter Wizard), or a filter with a less precise response but which allows a cheaper solution (Filter Design Tool and Analog Filter Wizard), choosing cheaper components (all but S/FILSYN) and/or with equal parameters (Filter Design Tool, FilterLab and Analog Filter Wizard).

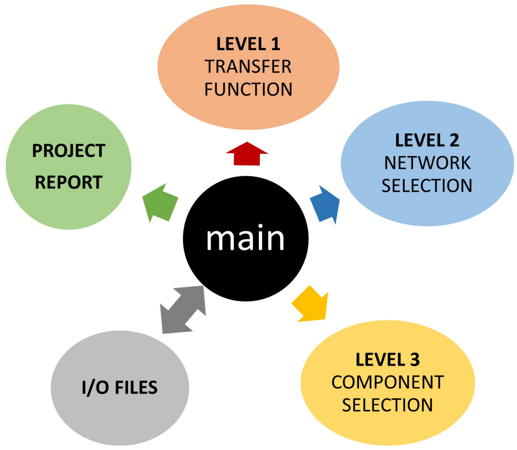


Fig. 1. General diagram of the tool developed.

Although providing an easy way to design a filter, all tools fail to provide a full customizable design where the user can control all the desired parameters of the filter. On the other hand, although an engineer specialized in the design and implementation of filters, knows exactly the solution he intends to implement, he will be able to benefit from a tool having plenty of options available to choose from, with several degrees of optimization. When considering the opposite case, a non-expert user, he will greatly benefit from a more automated process, where some simple specifications must be provided, and the optimization choices are made by the tool that will provide the final solution at the circuit level, with the list of components and respective advisable tolerances, as well as the OpAmp model. Another aspect that these tools fail to provide is the transfer function of the filter or sections, that are useful for post-processing, allowing to evaluate aspects of the frequency response and the time domain response, according to their importance for each filter application. Mainly, the tools are focused on providing solutions at circuit level, preferably with standard components sold by the company they are associated with, thus restricting degrees of freedom to the designer. It is therefore of interest to develop algorithms, and software tools based on those algorithms, in order to increase flexibility, automation and optimization of the filter design and implementation process, to bring tools to a full real-life context, taking into account the availability of components and their cost/benefit relation in order to meet a client specifications and expectations.

IV. Developed filter design and implementation tool

The developed tool is programmed in a structure of blocks controlled by the **main** block, as represented in Fig. 1.

The **I/O files** block contains the input and output files of the tool. The input files are the specifications of the project, the default definitions (for instance, the weights to consider in each cost function to evaluate) and data files (for instance, the normalized prototype filter tables [14] and the E series standard component values [6]). The output files are the tables and figures with the information about the implemented project

TABLE I
CRITERIA OF CHOICE OF AN OPTIMIZED SOLUTION IN LEVEL 1.

Criterion	Default weight	LP/HP	BP
Order	50%	$1 - \dfrac{N_x - N_{MIN}}{N_{MAX} - N_{MIN}}$	
Frequency	30%	$\dfrac{1}{1 + \lvert f_{Sx} - f_{Sspec}\rvert}$	$\max \left[\dfrac{1}{1 + \lvert f_{S1,2x} - f_{Sspec}\rvert} \right]$ OR $\dfrac{1}{1 + \lvert b_{Sx} - b_{Sspec}\rvert}$
Attenuation	20%	$\dfrac{A_{Sx} - A_{SMIN}}{A_{SMAX} - A_{SMIN}}$	

on the different levels. The **PROJECT REPORT** block assembles the output files generating *LaTeX* files with the information (available to be post-processed) and automatically generates a PDF report. From now on, the **Client** is the **User** interacting with the tool at the different level.

The developed tool has the capability of processing lowpass (LP), highpass (HP) and bandpass (BP) polynomial filters. The default mode of the developed tool, named **automatic mode**, tackles three of the main disadvantages found in the tools discussed in Sec. III — the lack of an automated process where the user plugs in a file with specifications and gets a possible optimized filter implementation; the lack of an explicit transfer function expression and the lack of the possibility of having components with different tolerances. In this mode, there are three levels of project to consider.

Level 1 concerns all the calculations, evaluation and optimized decisions to obtain the filter transfer function. The goal is to transform the **specifications** of a **client** into a filter transfer function $T(s)$, that establishes the relation between the output and the input of the filter, characterizing it [1] [4].

Because an ideal filter is never achievable, approximation methods need to be considered in order to arrive to a transfer function that fulfills the specifications. Different approximation methods are optimized for certain desired characteristics: having a constant gain in the passband, having a steeper transition in the transition band, having a constant group-delay in the passband, etc. [1] [4]. Depending on the application, one approximation will be more suitable than all the others, allowing a certain type and degree of optimization. Here, the classical polynomial approximations — Butterworth Chebyshev and Bessel (only for LP filters) — are considered. The approximation, or response shape, can either be specified in the input file or, otherwise, both the Butterworth and Chebyshev approximations are considered and processed individually, and then compared so the best solution is chosen.

The flow to obtain the transfer function starts with specifications that are converted into LP prototype specifications that, using the available filter tables files [14], allow to obtain a LP prototype transfer function that, using the frequency transformations, allow to obtain the desired transfer function [15] [16]. The use of a LP prototype and the frequency transformations constrains the design of BP filters to even order. The first type of specifications to consider are passband specifications. For Butterworth and Chebyshev filters these specifications concern the gain, the frequency limit(s) and the ripple in the magnitude response of the filter. For Bessel filters, the specification is the DC group-delay of the filter. Then, either the order of the filter (N) or a set of stopband specifications need to be considered. For Bessel filters, it is mandatory to specify N. For Butterworth and Chebyshev filters, the stopband specifications concern the stopband frequency limit(s) and the minimum attenuation to guarantee at those limits. These specifications allow to compute the order of the filter to implement through explicit equations [15] [1] that allow to obtain the correct order N.

For the Butterworth filter, the correct order N is the nearest integer above the computed value. However, there can be the case where the computed value is so close to the nearest integer below the computed value that both values for the order shall be considered — for instance, if the computed value is 4.1 the possible orders may be 4 and 5. Of course, with order 4 the specifications will not exactly be met, but, as an inferior order can mean less components (in this example, one less section, considering the cascaded active realization), only failing the specifications by a small margin might be worth it. For Butterworth filters, the below order is up to consideration if the decimal part of the computer number is ≤ 0.25. For Chebyshev filters, an odd and even order are computed for the different A_S obtained, both values are rounded to the next odd or even integer, and the smallest between the two is considered. Analogous to the Butterworth case, if the difference between the computer number and the below odd/even order is now ≤ 0.5, than that order might be up for consideration: for instance, if the computed odd order is 2.3 and the computed even order is 2.2, it results in a correct odd order of 3 and a correct even order of 4, respectively, and the smallest order of 3 is picked. But because $|2.2 - 2| = 0.2 < 0.5$, a filter of order 2 might also be considered (noticing that as $|2.3 - 1| = 1.3 > 0.5$, the below odd order is not up for consideration).

If more than one transfer function is obtained (either if the Butterworth and the Chebyshev solutions are compared, or there is more than one available order for each of these approximations), a first optimization decision needs to be taken. For that, a cost function related to the filter characteristics in the stopband is considered. There are three criteria that are considered — the order (F_{order}), the stopband frequency ($F_{frequency}$) and the stopband attenuation ($F_{attenuation}$). The criteria, respective functions for the different types of filters and default weights (an input file, changeable by the user) are indicated in Tab. I. With all these cost functions computed, the global cost function for a solution x is computed as $F(x) = w_{order} \times F_{order} + w_{frequency} \times F_{frequency} + w_{attenuation} \times F_{attenuation}$, producing a global value in the $[0,1]$ interval of the merit of this solution (where 1 is the best and 0 is "meritless"). The solution closest to 1 is picked as the optimal one.

Level 2 concerns the network selection to design the filter circuit, either in passive or active **technology**. In the passive technology, the LC ladder structures are considered. In the active technology, the 1st and 2nd-order sections are considered, to be used in a cascaded realization. By default, if the passband gain of the filter is less than or equal to 1, the passive technology is selected, due to their lower sensitivities and absence of the active element. Otherwise, the active technology is used. However, the technology choice can also be a specification defined by the **user**, in one of the input definition files (**I/O Files**). Different technologies offer advantages and disadvantages. For instance, in applications such as anti-aliasing lowpass filtering it is key to guarantee a good performance at high frequencies, where a choice of active filters can be a problem, while passive filters may offer better solutions [1]. On the other hand, the presence of inductors in passive filters also might give rise to issues due to their size and their electromagnetic interference [1] [17].

For LC ladder networks, two dual structures are considered — the T structure and the π structure. By default, the T structure is selected, however, like the technology, the choice between the T and π structures can be made by the **Client** in the **I/O Files**. Regarding the resistive terminations, the network to consider is always doubly-terminated by R_{in} and R_{out}, except for the case of unitary-gain, where it is single terminated at the output by R_{out}. The series/parallel reactive elements for the LP, HP and BP cases are indicated in [18] [16] [1]. To obtain the transfer function of the LC ladder network, the LC elements and its resistive terminations are considered as a sequence of cascaded two-port networks, each described by a transmission matrix.

For the active technology, the first step consists on factorizing the global transfer function into sections, that are ordered and have their gain assigned in an optimized way. Then, for each section, the (K, Q) parameters of their transfer function allow the optimal and automatic selection of the network. The optimized factorization of the transfer function takes two criteria into account — the sections ordering and the gain assignment to each section. First, the global transfer function denominator is decomposed in 2nd-order polynomials (with a 1st-order term for odd order in the LP and HP cases). Each of these polynomials allows to extract the (ω_p, Q) parameters of each section and order them from the lowest to the highest Q, with the eventual 1st-order section being placed in first place. With the ordered sections, the gain is assigned as described by $K_i = \begin{cases} K_{global} \times \dfrac{M_{\lfloor \frac{N+1}{2} \rfloor}}{M_1} & \text{for } i = 1 \\ \dfrac{M_{i-1}}{M_i} & \text{for } i = 2, ..., \left\lfloor \dfrac{N+1}{2} \right\rfloor \end{cases}$ with

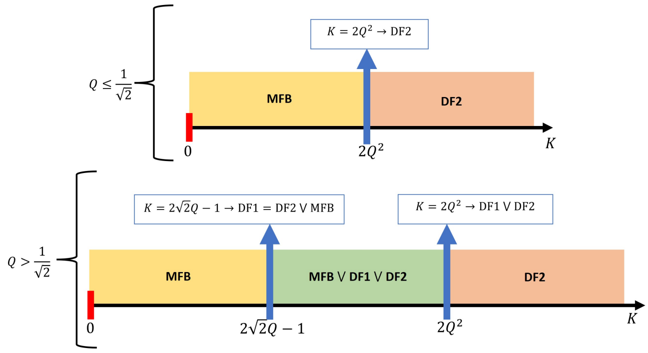


Fig. 2. Regions of validity for the different BP biquadratic sections design.

$$K_{global} = \begin{cases} K & \text{for LP and HP filters} \\ \dfrac{a_{N/2}}{\prod_{i=1}^{N/2} \dfrac{\omega_i}{Q_i}} & \text{for BP filters} \end{cases}.$$

For the 1st-order LP and HP sections, the value of K determines the network to choose — for $K < 1$ the inverting structure is selected, while for $K = 1$ the non-inverting structure is preferred. For $K > 1$, different criteria need to be evaluated in order to decide between the two possible networks. These criteria are the number of components; the value of the component spread; the passive sensitivities, measured by the worst-case index I_W referred for K, $I_W(K) = \sum_{m=1}^{M} |S_{x_m}^{K}|$; and the active sensitivity, measured by ω_p-to-GBW sensitivity normalized to the ω_p/GBW ratio. The default weights for each criteria are, respectively, 10%, 20%, 50% and 20%. They are also an input file from the **I/O files** block, and the **user** can change these weights to better fit his project. It is verified that only for the number of components criterion the inverting structure has preference over the non-inverting structure and, therefore, only if the weight of this parameter is modified to be above 50%, this structure is selected.

Similarly, for 2nd-order LP and HP sections, the multiple-feedback (MFB) structure should be selected for $K < 1$ and the Sallen-Key (SK) structure should be selected for $K = 1$. For $K > 1$ the decision between the two networks is, again, based through the analysis of different criteria. The criteria are same as the 1st-order case, with the modification of considering the passive sensitivity through two criteria instead of one: $I_W(K)$ and I_W referred to Q, $I_W(Q) = \sum_{m=1}^{M} |S_{x_m}^{Q}|$. In the 2nd-order case, as the decision is not binary for all the criteria as it was for the 1st-order case, cost functions are considered for some of them. For the component spread, the goal is to obtain a component spread close to 1 (equal components). Therefore, the worst component ratio is determined for each structure, and the cost function measures how close they are to 1, $F_{spread}(x) = \dfrac{1}{1 + |x - 1|}$. As for the sensitivities, the cost function translates the goal of minimizing them — for each case, x_{MIN} will correspond to the minimum sensitivity index computed between the two structures and x_{MAX} will correspond to the maximum, resulting in the function $F_{sensitivity}(x) = 1 - \frac{x - x_{MIN}}{x_{MAX} - x_{MIN}}$ Therefore, the structure with x_{MIN} will result in a 1 cost function value ("full merit"), while the structure with x_{MAX} will correspond to a 0 cost function value ("meritless").

For the BP case, two to three solutions can coexist at the same time. In this case the minimum number of component achievable is 4 with the Deliyannis-Friend (DF) structure when $\alpha_{DF} = 1$ (the non-inverting input of the OpAmp connected to ground) and, therefore, that is the value that the cost function needs to approximate, $F_{components}(x) = \frac{1}{1 + |x - 4|}$ Fig. 2 summarizes the different regions for Q and K where every design (MFB, DF solution with β_1 (DF1) or DF solution with β_2 (DF2)) is valid. For $Q \leq \frac{1}{\sqrt{2}}$, the DF2 solution is considered for $K \geq 2Q^2$ and the MFB solution is considered otherwise. For $Q > \frac{1}{\sqrt{2}}$: for $0 < K < 2\sqrt{2}Q - 1$ only the MFB solution is valid; for $2\sqrt{2}Q - 1 \leq K < 2Q^2$, all three solutions are valid (for $K = 2\sqrt{2}Q - 1$, DF1 and DF1 are equal, so it is enough to consider one of them); for $K = 2Q^2$ DF1 and DF2 are both valid solutions and, for $K > 2Q^2$ only DF2 is a solution to consider.

Level 3 concerns the component selection to design and implement the filter circuit. This level maps the chosen network of the filter circuit with the filter transfer function, effectively designing and implementing the filter circuit. The aspect of considering the component series is very important because discrete electronic components do not exist in a continuum interval of values. Ideal OpAmps also do not exist. In fact, while often the project is done considering the existence of an infinite possibility values for component parameters, as well as an ideal active element, then, when the task of picking real value components arrives, the expected circuit behaviour deviates from what was originally designed. In an optimization process the fact that these real components produce a deviation needs to be taken into account. So, at this level, first the ideal passive components are designed, and then those ideal values are converted to the real values from the E series, and the minimum GBW of the OpAmp is determined.

Starting with the ideal component design, the method for designing the LC Ladder structures is using the coefficient matching technique. An additional input specification for the passive implementation is the output resistance R_{out}. For active filters, each section is individually designed using the design methods, such as described in [17] [1] [19] [20].

The next step is to match to select the standard available components to optimally implement the filter. For the passive components, their ideal value needs to be converter to the real values from the standard E series, and the respective tolerance needs to be picked (see Tab. II). As it was verified in [8], more than the tolerance of the components, deviations from the ideal values by using standard available values cause deformations in the filter response. The ideal value is considered with a mantissa and an exponent. The mantissa is considered in the $[1, 10[$ interval, as this is the scale of values of the different E series [6]. Then, by default, it is considered that a deviation higher than 0.05 between the ideal value mantissa and the closest value available in a specific Exx series is enough to discard that series. This default value is also on an input file (**I/O files**) changeable by the **user**. For passive filters, their lower sensitivities allow a reasonable search starting from series E6 (20% of tolerance) down to series E192 (0.5% of tolerance). However, for active sections, due to their higher sensitivities, the optimal search starts with series E24 (5% of tolerance) [20] [19]. However, this can also be a definition changeable by the **user** in the **I/O Files**. Finally, the minimum GBW of the OpAmp is calculated as 2 decades above the frequency limit of the passband (higher frequency edge for BP filters). Because for HP filters, there might be the need to further extend the OpAmp bandwidth, this definition is also changeable by the **user** on the **I/O Files**. Finally, the minimum GBW of the OpAmp is calculated as 2 decades above the frequency limit of the passband.

TABLE II
AVAILABLE SERIES OF STANDARD COMPONENTS AND RESPECTIVE TOLERANCE.

E Series	Tolerance
E24	5%
E48	2%
E96	1%
E192	0.5%

As discussed before, another limitation of the available tools is that it lacks some customization of the filter parameters that might be useful for an expert **user** that already knows with more precision the response characteristics of the filter he wants to implement. With that in mind a second mode, an **expert mode**, was developed. This mode is aimed specifically to the implementation of active filters in cascaded sections, by specifying the type and the (K, f_p, Q) parameters of each section. Unlike the **automatic mode**, the **expert mode** allows the implementation of odd order BP filters, as well as the use LP and HP sections to implement them. The parameters of each section can either be directly specified, or be calculation through other specified parameters, such as the DC group-delay τ_0, the poles frequency group-delay τ_p, the bandwidth b_P or the maximum of the response (f_{max}, K_{max}). Regarding the project flow, the **expert mode** skips **Level 1** entirely as the transfer function is not determined through the classical approximations, and, therefore, it also skips all the factorization, ordering and gain assignment steps in **Level 2**, going directly into the automatic and optimized selection of the network to design and implement each section.

V. BANDPASS FILTER PROJECT — AN EXAMPLE

In this section, an example of a bandpass filter project with an automated and optimized solution is presented, allowing to illustrate some of the features of the developed tool. A **Client** needs a solution for a radio navigation **Application** in his company. The client defines a set of **Specifications**: it is a maximally flat (Butterworth) bandpass filter that must guarantee a gain $G_P = 6\,\text{dB}$, the passband limits are $f_{P1} = 4\,\text{kHz}$ and $f_{P2} = 16\,\text{kHz}$, and the maximum ripple in the passband shall be $A_P = 0.5\,\text{dB}$. The filter must guarantee a minimum attenuation $A_{min1} = 35\,\text{dB}$ at $f_{S1} = 1\,\text{kHz}$ and $A_{min2} = 40\,\text{dB}$ at $f_{S2} = 50\,\text{kHz}$.

In **Level 1**, using the Butterworth approximation and the specifications given, the tool optimally and unequivocally determines a 10th-order transfer function from a 5th-order prototype. Fig. 3 shows the results at this level, presenting a screenshot of the report.

As the filter provides gain, the **technology** to use is active filters. Therefore, at **level 2**, the first step is to factorize the transfer function and optimally order and determine the gain of each section. The screenshot of the report, shown in Fig. 4, shows a table with all the sections information and a figure with their individual and accumulated response. The next step is to select the network for each section. The tool automatically and optimally (through the cost function, when it is necessary, such as Section E) determines each section: sections A and C will be designed using the DF2 solution, sections B, D and E will be designed using the MFB solution. The network for each section is also presented in Fig. 4.

Following the choice of the network for each section, the next step is the component selection in **Level 3**. The report screenshot of Fig. 5 illustrates the table with the results for this level with the component designation for each section, the respective ideal and real values and the picked series and respective tolerance. It also shows the minimum GBW of the OpAmp to use. As it is visible, for some components, only the tightest tolerance series, E192, guarantees that there is not much deviation between the ideal value and the real value of the component, for example, for R_{2A} and R_{2E}.

The project is concluded and the **Client** now has a full characterization of his optimized solution: the transfer function, the networks to implement the filter circuit, and the component real values. Next, the circuit is simulated to observe the solution. To do so, the LT1007 [21] is selected, as it has a $\text{GBW}/2\pi$ of $8\,\text{MHz}$ (so, it fulfills the project specification of $\geq 1.6\,\text{MHz}$). The simulation results are presented in Fig. 6. In blue, the response of the filter with the real components is indicated showing that it fulfills the specifications. For instance, it is verified that at $f_{S1} = 1\,\text{kHz}$ the gain is $G\left(f_{S1}\right) = -55.9\,\text{dB}$ and at $f_{S2} = 50\,\text{kHz}$ the gain is $G\left(f_{S2}\right) = -46.1\,\text{dB}$. In red, the response of the filter with the ideal passive component values is indicated and, in purple, the ideal transfer function response is shown. It is verified that the three responses are practically identical, validating the optimization both in the ideal design, and in the implementation with the selected components. In gray, a statistical Monte Carlo [22] analysis, with 100 runs and gaussian distribution for the component values is presented. It is verified that the maximum gain deviation is around $4\,\text{dB}$ in the passband and the maximum group-delay deviation is around $15\,\mu\text{s}$ at the peak.

Next, the optimized solution obtained from the developed tool is compared with the solution obtained with the Analog Filter Wizard tool for the same project. The results are indicated in Fig. 7, showing several differences – regarding the topology, the Analog Filter Wizard uses three DF (although they interchangeably call it multiple-feedback) and two MFB sections, while it is the other way around for the presented solution. The sections ordering and gain splitting is also done differently. A major difference is that while the implemented tool independently chooses a series and respective tolerance for the passive components, the Analog Filter Wizard only allows to define a global tolerance/series for the resistors and for the capacitors. All these choices result in designs with bigger deviation margins both for the gain (around $6\,\text{dB}$ in the passband) and for the group-delay (around $75\,\mu\text{s}$ in the peak), showing that the design produced by the developed tool in this work is more optimized in the sense of minimizing these margins.

The developed tool proves its usefulness on increasing the flexibility, automation and optimization of the filter design and implementation process to obtain a solution for this project.

VI. Conclusions

In this work, a tool was developed that allows the optimized and automated design and implementation of filters from a set of specifications. Currently, the tool developed defines the filter optimization application flow, which is independent of the form or support application. In the future, an application may be implemented based on any platform with a development environment and a certain graphical interface.

The tool showed the existence of different levels of optimization, with different criteria and associated cost functions, enforcing the MOO nature of the problem. It allows to implement filters with polynomial characteristics. The optimization was adapted in the different steps of the project: in **level 1**, the filter response (in frequency and/or time) is optimized according to the specifications, establishing the response shape (Butterworth, Chebyshev or Bessel), the order of the filter and having as output the transfer function of the filter $T(s)$. **Level 2** deals with optimization in terms of technology (passive or active), topology and circuit structure (ladder LC for passive filters, and cascaded sections for active filters). The implementation of cascaded sections also implies optimization in terms of factoring, ordering and assigning the gain of each section. Its output is the circuit to be designed. **Level 3** deals with optimizing the design of the circuit to be implemented, establishing the parameters of the ideal components. Finally, optimization is carried out at the level of matching the ideal value of the component with a real value available in standard components and with an associated tolerance. It thus establishes the parameters of the real components (value and tolerance) for the practical implementation of the circuit with standard available components. It also determines the minimum GBW of the OpAmp to use. The final output of the tool is a PDF **project report** with all the optimized solution information to implement the filter. Besides the **automatic mode** just described, with a classic definition of the specifications, the tool also allows an **expert mode**, in which each section of the filter is well defined through its parameters. The existence of this mode allows, for example, to define odd-order BP filters by combining first-order LP and/or HP sections with biquadratic BP sections.

In summary, the developed tool proves its usefulness on increasing the flexibility, automation and optimization of the

LEVEL 1 - TRANSFER FUNCTION

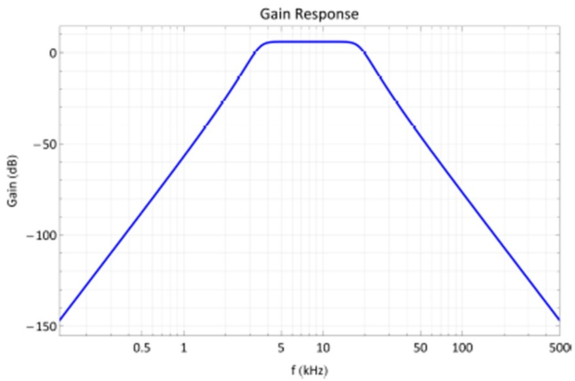
Table 1: Filter specifications.

G_P	A_P	f_{P1}	f_{P2}	A_{min1}	f_{S1}	A_{min2}	f_{S2}
6 dB	0.5 dB	4 kHz	16 kHz	35 dB	1 kHz	40 dB	50 kHz

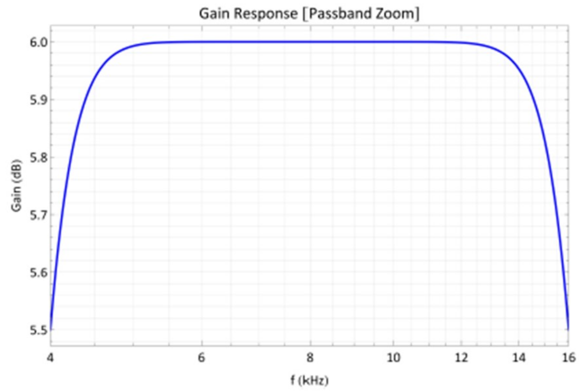
$$T(s) = \frac{N(s)}{D(s)} = \frac{a_5 \times s^5}{\left(\sum_{i=0}^{9} b_i \times s^i\right) + s^{10}}$$

Table 2: Transfer function coefficients.

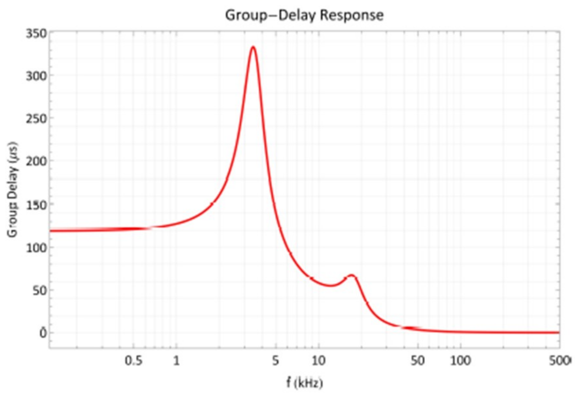
a_5	1.39185×10^{25}	b_0	1.02967×10^{47}
b_1	1.22714×10^{43}	b_2	9.35007×10^{38}
b_3	4.63577×10^{34}	b_4	1.6425×10^{30}
b_5	3.98267×10^{25}	b_6	6.50078×10^{20}
b_7	7.26177×10^{15}	b_8	5.7969×10^{10}
b_9	3.01118×10^{5}		



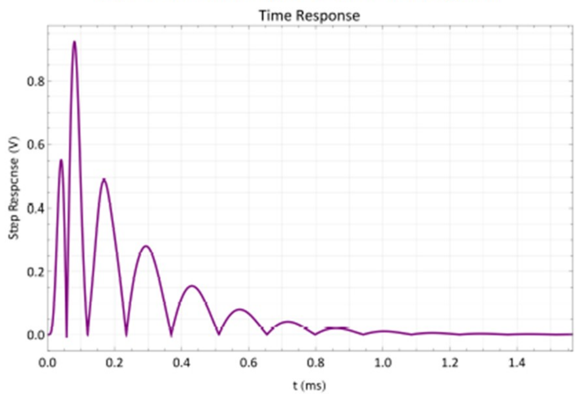
(a) Gain.



(b) Zoom on the passband of the gain.



(c) Group-delay.



(d) Step response.

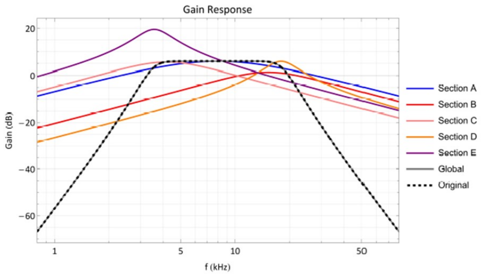
Figure 1: Filter response.

Fig. 3. Project report for Level 1.

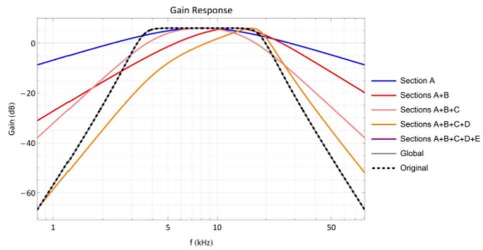
LEVEL 2 - NETWORK SELECTION

Table 3: Sections transfer function and parameters.

Section	$T(s)$	K	ω_p	Q
A	$\dfrac{(1.8566 \times 10^5)\, s}{s^2 + (9.30505 \times 10^4)\, s + 2.52662 \times 10^9}$	1.99526	$50.2655\,\mathrm{krad\,s^{-1}}$	0.540196
B	$\dfrac{(1.38688 \times 10^5)\, s}{s^2 + (1.18063 \times 10^5)\, s + 9.17964 \times 10^9}$	1.1747	$95.8104\,\mathrm{krad\,s^{-1}}$	0.81152
C	$\dfrac{(6.24306 \times 10^4)\, s}{s^2 + (3.24958 \times 10^4)\, s + 6.9543 \times 10^8}$	1.92119	$26.371\,\mathrm{krad\,s^{-1}}$	0.81152
D	$\dfrac{(9.53862 \times 10^4)\, s}{s^2 + (4.79999 \times 10^4)\, s + 1.27547 \times 10^{10}}$	1.98722	$112.937\,\mathrm{krad\,s^{-1}}$	2.35285
E	$\dfrac{(9.0772 \times 10^4)\, s}{s^2 + (9.50844 \times 10^3)\, s + 5.00505 \times 10^8}$	9.54646	$22.372\,\mathrm{krad\,s^{-1}}$	2.35285

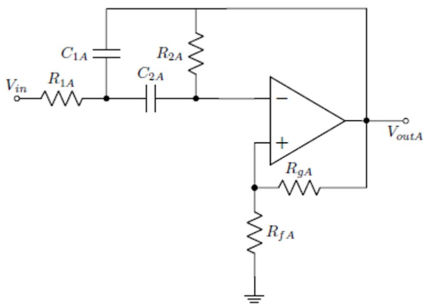


(a) Individual response of the sections.

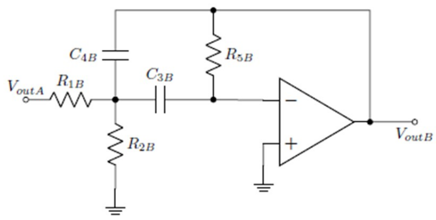


(b) Accumulated response of the sections.

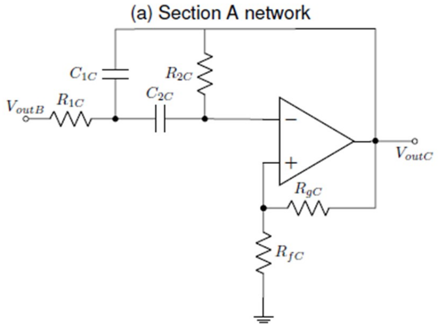
Figure 2: Sections response.



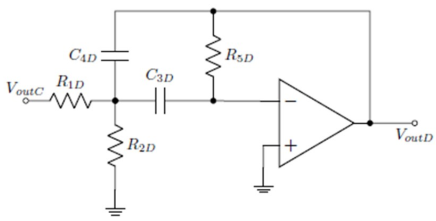
(a) Section A network



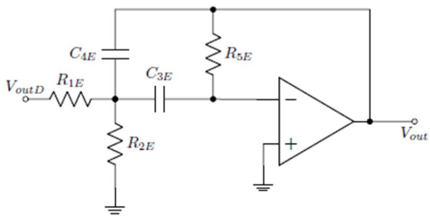
(b) Section B network



(c) Section C network



(d) Section D network



(e) Section E network

Figure 3: Network selection.

Fig. 4. Project report for Level 2.

LEVEL 3 - COMPONENT SELECTION

Table 4: Circuit design and tolerance design.

Component	Ideal	Series	Real	Tolerance
SECTION A				
R_{1A}	$51.2971\,\mathrm{k\Omega}$	E24	$51\,\mathrm{k\Omega}$	5%
C_{1A}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
R_{2A}	$7.715\,57\,\mathrm{k\Omega}$	E96	$7.68\,\mathrm{k\Omega}$	1%
C_{2A}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
R_{gA}	$7.715\,57\,\mathrm{k\Omega}$	E96	$7.68\,\mathrm{k\Omega}$	1%
R_{fA}	$65.7661\,\mathrm{k\Omega}$	E192	$65.7\,\mathrm{k\Omega}$	0.5%
SECTION B				
R_{1B}	$7.210\,42\,\mathrm{k\Omega}$	E192	$7.23\,\mathrm{k\Omega}$	0.5%
R_{2B}	$59.4677\,\mathrm{k\Omega}$	E48	$59\,\mathrm{k\Omega}$	2%
C_{3B}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
C_{4B}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
R_{5B}	$16.9401\,\mathrm{k\Omega}$	E48	$16.9\,\mathrm{k\Omega}$	2%
SECTION C				
R_{1C}	$55.2329\,\mathrm{k\Omega}$	E96	$54.9\,\mathrm{k\Omega}$	1%
C_{1C}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
R_{2C}	$26.0345\,\mathrm{k\Omega}$	E48	$26.1\,\mathrm{k\Omega}$	2%
C_{2C}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
R_{gC}	$63.7381\,\mathrm{k\Omega}$	E48	$63.4\,\mathrm{k\Omega}$	2%
R_{fC}	$26.0345\,\mathrm{k\Omega}$	E96	$26.1\,\mathrm{k\Omega}$	1%
SECTION D				
R_{1D}	$10.4837\,\mathrm{k\Omega}$	E24	$10\,\mathrm{k\Omega}$	5%
R_{2D}	$2.293\,26\,\mathrm{k\Omega}$	E48	$2.26\,\mathrm{k\Omega}$	2%
C_{3D}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
C_{4D}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
R_{5D}	$41.6667\,\mathrm{k\Omega}$	E96	$41.2\,\mathrm{k\Omega}$	1%
SECTION E				
R_{1E}	$11.0166\,\mathrm{k\Omega}$	E24	$11\,\mathrm{k\Omega}$	5%
R_{2E}	$68.9465\,\mathrm{k\Omega}$	E192	$69\,\mathrm{k\Omega}$	0.5%
C_{3E}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
C_{4E}	$1\,\mathrm{nF}$	E24	$1\,\mathrm{nF}$	5%
R_{5E}	$210.339\,\mathrm{k\Omega}$	E48	$215\,\mathrm{k\Omega}$	2%

OPAMP: $GBW \geq 1.6\,\mathrm{MHz}$

Fig. 5. Project report for Level 3.

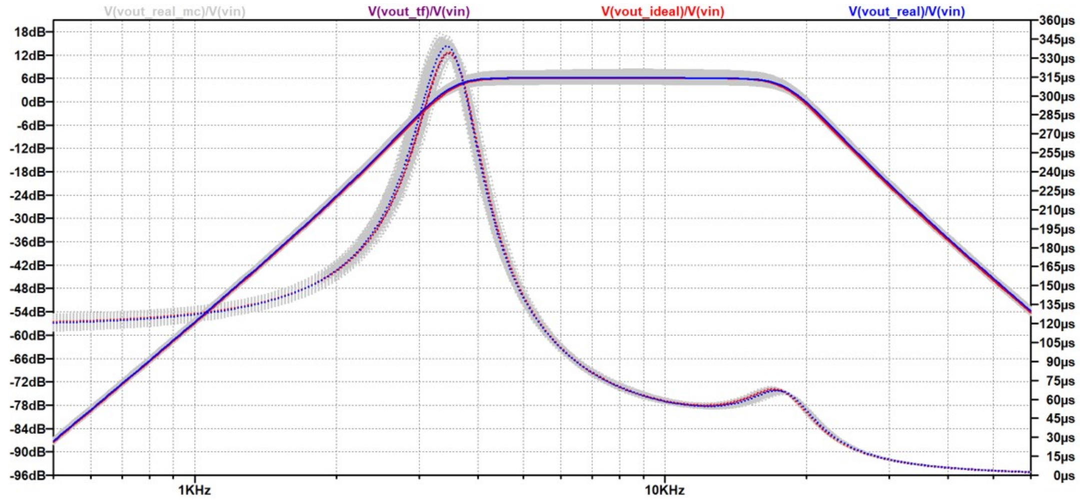


Fig. 6. Bandpass filter project gain and group-delay: real circuit with standard components (in blue), ideal circuit (in red), transfer function (in purple) and Monte Carlo analysis (in gray).

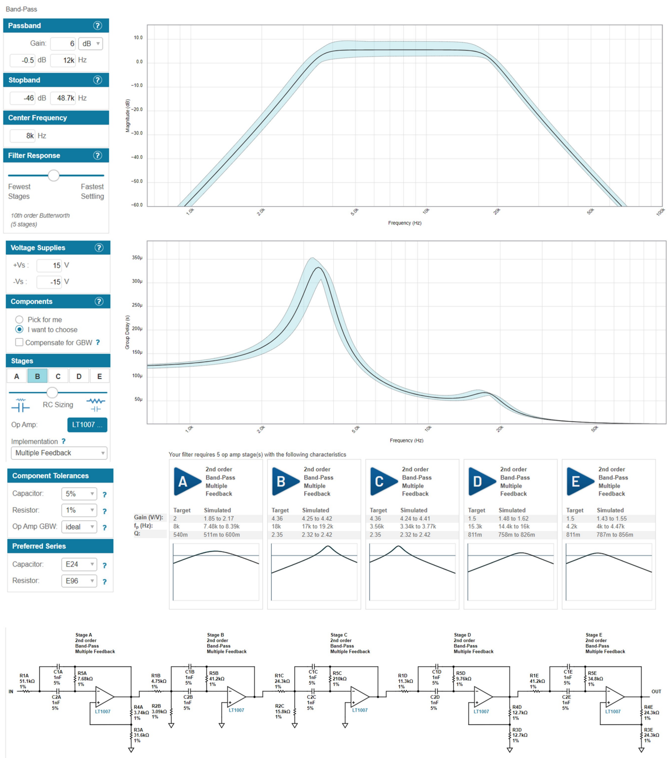


Fig. 7. Solution obtained using Analog Filter Wizard tool for the bandpass project example.

filter design and implementation process, being useful both for specialized engineers that have more options and degrees of optimization, and non-expert users that can benefit from the automation of the various steps of the process, from the transfer function until the circuit implementation.

REFERENCES

[1] M. A. Siddiqi, *Continuous Time Active Analog Filters*, 1^{st} ed. Cambridge University Press, 2020, iSBN:978-1-108-48683-5.

[2] A. I. Zverev, *Handbook of Filter Synthesis*, 1^{st} ed. John Wiley and Sons, Inc., 1967.

[3] G. Daryanani, *Principles of Active Network Synthesis and Design*, 1^{st} ed. Wiley, 1991.

[4] T. M. de Almeida, *Biquadratic Sections, Slides of the "Analog and Digital Filters" Course*. Instituto Superior Técnico, 2020/2021.

[5] Y. C. Kuyu and F. Vatansever, "Analog filter group delay optimization using metaheuristic algorithms: A comparative study," *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, Sep. 2018, doi:10.1109/IDAP.2018.8620754.

[6] ElectronicsNotes, "Standard resistor values," January 2022, accessed at 15/01/2022.

[7] O. B. Kchaou, A. Sallem, P. Pereira, M. Fakhfakh, and M. H. Fino, "Multi-objective sensitivity-based optimization of analog circuits exploiting nsga-ii front ranking," *2015 International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*, Sep. 2015, doi:10.1109/SMACD.2015.7301696.

[8] J. P. Azevedo, *Design and Implementation of Optimized Filters - 2nd Cycle Integrated Project in Electrical and Computer Engineering*. Instituto Superior Técnico, 2021/2022.

[9] A. Engineering, "S/filsyn," September 2022, accessed at 09/10/2022.

[10] Panasonic, "Lc filter simulator," September 2022, accessed at 09/10/2022.

[11] T. Instruments, "Filter design tool," September 2022, accessed at 09/10/2022.

[12] Microchip, "Filterlab," September 2022, accessed at 09/10/2022.

[13] A. Devices, "Analog filter wizard," September 2022, accessed at 09/10/2022.

[14] T. M. de Almeida, *Filter Design Tables - "Analog and Digital Filters" Course*. Instituto Superior Técnico, 2020/2021.

[15] ——, *Approximation Methods, Slides of the "Analog and Digital Filters" Course*. Instituto Superior Técnico, 2020/2021.

[16] ——, *Frequency and Network Transformations, Slides of the "Analog and Digital Filters" Course*. Instituto Superior Técnico, 2020/2021.

[17] ——, *Active-RC Filters, Slides of the "Analog and Digital Filters" Course*. Instituto Superior Técnico, 2020/2021.

[18] ——, *LC Ladder Filters, Slides of the "Analog and Digital Filters" Course*. Instituto Superior Técnico, 2020/2021.

[19] H. Zumbahlen, *Basic Linear Design*, 1^{st} ed. Analog Devices, 2007.

[20] B. C. Baker, *ADN003: Select the Right Operational Amplifier for your Filtering Circuits*. Microchip Technology Inc., 2003.

[21] *LT1007 op-amp datasheet*, Linear Technology, 2015. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/LT1007-LT1037.pdf

[22] D. C. Montgomery and G. C. Runger, *Applied Statistics and Probability for Engineers*, 7^{th} ed. Wiley, 2018.