Numerical Methods for the Robust Stability and Performance of Power Conversion Systems

Ramon Estalella-Rodríguez, Carlos Olalla, Isabelle Queinnec, Angel Cid-Pastor

Numerical Methods for the Robust Stability and Performance of Power Conversion Systems

Ramon Estalella-Rodríguez
Dept. d'Enginyeria Electrònica, Elèctrica i Automàtica
Universitat Rovira i Virgili
Tarragona, Spain
ramon.estalella@urv.cat

Carlos Olalla
Dept. d'Enginyeria Electrònica, Elèctrica i Automàtica
Universitat Rovira i Virgili
Tarragona, Spain
carlos.olalla@urv.cat

Angel Cid-Pastor
Dept. d'Enginyeria Electrònica, Elèctrica i Automàtica
Universitat Rovira i Virgili
Tarragona, Spain
angel.cid@urv.cat

Isabelle Queinnec
LAAS - CNRS
Université de Toulouse
Toulouse, France
queinnec@laas.fr

***Abstract*—One of the open problems in the design of power converters is related to the selection of components, and the effects that these choices may have, beyond steady-state properties, in the dynamical behaviour of the system. This paper describes a numerical method that may help in finding appropriate components. The method is based on Lyapunov functions that are efficiently manipulated to derive linear matrix inequality (LMI) based conditions. With the proposed approach, the domain of component values that verify given dynamical properties can be estimated. The resulting parametric domain can then be used by the practicing engineer to optimise other factors, such as weight, cost or volume. The advantages of the proposed approach have been illustrated with the design of two different input filters for power converters operating as constant power loads.**

I. Introduction

The design of power filters is an important part of the design of switching power converters. These converters often require input filters to reduce current and voltage ripple and to comply with conducted electromagnetic interference (EMI) regulations. Since input filters are typically designed on top of existing converters, designers must consider the possible interactions that may arise. Such interactions can not only make a switching converter lose some of the dynamic performance but even make it unstable.

The basis for what can be considered a conventional design can be seen on [1]–[3]. This method allows to select the parameters of the filter by using simplifications of the Nyquist stability criteria. Although the method is straightforward to apply, the simplifications might lead to conservative results.

This paper takes another route by using Lyapunov-based methods. Such approaches allow to ensure stability and performance properties of a dynamical system by solving linear matrix inequalities (LMIs). Such numerical tools should help to find regions in the space of the aforementioned parameters of the filter, ensuring design specifications. These specifications may include attenuation, stability, decay rate or damping ratio, to name some. Within the resulting region of parameters, the designer can choose appropriate components, according to other restrictions such as weight, cost, volume, fitting to commercial values, etc.

The paper follows the ideas in [4] to manipulate parameter-dependent Lyapunov functions and derives some original LMI conditions that are useful in the definition of the constraints. Although the proposed methods could be used to design different parts of a power converter, these ideas are illustrated with two examples of input filter design for power conversion systems behaving as constant power loads.

The paper is organized as follows. First, the basis for the robust stability and performance is introduced in Section II. Section III describes a bisection-based method to enlarge the parametric space that satisfies the stability and performance constraints. Section IV shows the case example of input filter design and evaluates the robustness of the proposed approach. Finally, conclusions are given in Section V.

II. Numerical Methods for Robust Stability and Performance

The investigation of the domain of stability that we propose is based on a convex combination of Lyapunov functions, following the ideas in [4]. Besides of stability, transient performance is addressed by robust pole placement.

A. Robust Stability

Consider the following continuous-time linear system

$$\dot{x}(t) = Ax(t) \tag{1}$$

It has been proven that A is asymptotically stable if and only if the LMIs

$$\mathbf{P} > 0; \quad A'\mathbf{P} + \mathbf{P}A < -\mathbf{I} \tag{2}$$

This work has been partially supported by project PID2020-120151RB-I00 financed by MCIN/ AEI /10.13039/501100011033. It has also been partially supported by University Rovira i Virgili through the Martí-Franquès Research grants Programme with reference 2019PMF-PIPF-26.

or equivalently, with $\mathbf{W} = \mathbf{P}^{-1}$

$$\mathbf{W} > 0; \quad A\mathbf{W} + \mathbf{W}A' < -\mathbf{I}, \tag{3}$$

admit a feasible solution. In (2) and (3), $\mathbf{I}$ is the identity matrix of appropriate dimensions and A' is the transpose of the state matrix A. The bold notation in $\mathbf{P}$ and $\mathbf{W}$ indicates that these are variables that must be found to prove stability.

In the case where A is not precisely known, and as long as the unknown parameters appear linearly, it is possible to define A as a polytope that contains the uncertain domain of parameters. The polytope can be modeled as the convex combination of its N vertices A_j as follows

$$A(p) = \sum_{j=1}^{N} \lambda_j A_j; \;\; \sum_{j=1}^{N} \lambda_j = 1; \;\; \lambda_j \geq 0. \tag{4}$$

where p represents the vector of parameters that are uncertain.

If there exists a common Lyapunov matrix $\mathbf{W}$ that verifies (3) for all the vertices of the polytope (4), the stability of the uncertain matrix A can be assured. Note that only by verification of the vertices, the properties hold in the entire domain of p. Also, this is only a sufficient condition for robust stability and such approach often leads to conservative results.

For less conservative results, the set of LMIs

$$\begin{aligned} &\mathbf{W}_j A_j' + A_j \mathbf{W}_j < -I; \quad j = 1, ..., N \\ &\mathbf{W}_k A_j' + A_j \mathbf{W}_k + \mathbf{W}_j A_k' + A_k \mathbf{W}_j < \frac{2}{N-1} I \\ &j = 1, ..., N-1; \quad k = j+1, ..., N \end{aligned} \tag{5}$$

may be used. In (5), $\mathbf{W}_j$ corresponds to a convex combination of positive definite matrices. The resulting Lyapunov function is defined as in (4), such that $\mathbf{W}(p)$ depends on p.

$$\mathbf{W}(p) = \sum_{j=1}^{N} \lambda_j \mathbf{W}_j; \;\; \sum_{j=1}^{N} \lambda_j = 1; \;\; \lambda_j \geq 0. \tag{6}$$

Proof of the sufficiency of these LMIs is given on [4]. The resulting parameter-dependent Lyapunov function provides some degree of flexibility, and therefore the results are less conservative than using a unique $\mathbf{W}$ for all vertices.

B. Robust Performance Constraints

The dynamical characteristics of the system to be designed can be specified by pole placement techniques. First, it is possible to relate some properties to a circle in the complex plane centered at the origin with radius ρ [5]. Other characteristics can be related to a conic sector in the left-half plane with an angle equal to θ [6]. Figure 1 shows these conditions in the complex plane.

1) Circle of Radius ρ: The maximum natural frequency of oscillation of the poles of a system can be constrained to ρ. The LMI conditions that verify the location of the system poles in a generic circular region have been derived in [5]. The LMIs for the particular case where the circle is centered at $0 + 0j$ are shown in Eqs. (7) to (9), where $A_{rj} = A_j - \rho\mathbf{I}$.

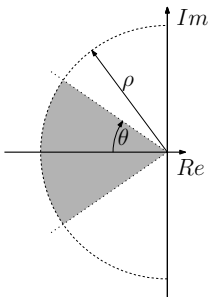


Fig. 1: Region defined with the pole placement LMIs.

2) Conic Sector of Angle θ: As shown in [6], the dynamic system (1) has its poles inside a conic sector of angle θ if and only if there exists $\mathbf{W} > 0$ such that

$$\begin{pmatrix} sin\theta(A\mathbf{W} + \mathbf{W}A') & cos\theta(A\mathbf{W} - \mathbf{W}A') \\ cos\theta(\mathbf{W}A' - A\mathbf{W}) & sin\theta(A\mathbf{W} + \mathbf{W}A') \end{pmatrix} < 0 \tag{10}$$

combining this LMI with the sufficient condition for parameter-dependent Lyapunov functions described in [4], two novel LMIs (11) and (12) have been derived. These LMIs consider parameter-dependent Lyapunov functions (6) and are one of the contributions of the paper.

C. Parameter Region

As stated in the introduction, the proposed algorithm does not return a single parameter value, but rather a region or a domain of values. As in [4], [5], consider the following continuous-time linear system:

$$\dot{x}(t) = \left(A_0 + \sum_{i=1}^{m} \alpha_i E_i \right) x(t) \tag{13}$$

where $x \in \mathbb{R}^n$, $A_0 \in \mathbb{R}^{n \times n}$ is the nominal matrix of the system, $E_i \in \mathbb{R}^{n \times n}$ $(i = 1, ..., m)$ are given matrices representing the perturbation directions and $\alpha_i \in \mathbb{R}$ $(i = 1, ..., m)$ are scalar values defining the amount of perturbations allowed.

The parameter dependent matrix in (13) is equivalent to the representation in (4). Note that the sum of A_0 with the different E_i matrices multiplied by the scalar values in the vector α_i results in the vertices of the domain of $A(p)$. It is worth to remark, however, that this approach allows to consider the influence of the parameters in p separately.

Each perturbation direction is related to a vector α_i, which contains two values (lower and upper) that are connected to the range of possible parameter values. Thus, the vector is defined as $\alpha_i := [-\underline{\alpha_i}, \; \overline{\alpha_i}]$. The aim is to maximise the domains $[-\underline{\alpha_i}, \; \overline{\alpha_i}]$ $\forall i = 1, \ldots, m$ while ensuring stability and pole placement conditions.

A stable initial A_0 is assumed. In the case where pole placement is required, also an initial A_0 with its poles inside the region is assumed.

The problem is written such that each perturbation direction can be related to an uncertain parameter of the system. The range of values for each parameter depends on: the initial values (set on A_0), the *scale* of the parameters (set on E_i) and

$$\begin{pmatrix} A_{rj}\mathbf{W}_j + \mathbf{W}_j A'_{rj} & A_{rj}\mathbf{W}_j \\ \mathbf{W}_j A'_{rj} & -\rho\mathbf{W}_j] \end{pmatrix} < \begin{pmatrix} -\mathbf{I} & 0 \\ 0 & 0 \end{pmatrix} \quad ; \quad j = 1, ..., N \tag{7}$$

$$\begin{pmatrix} A_{rj}\mathbf{W}_j + \mathbf{W}_j A'_{rj} + A_{rj}\mathbf{W}_k \\ +A_{rk}\mathbf{W}_j + \mathbf{W}_j A'_{rk} + \mathbf{W}_k A'_{rj} & A_{rj}\mathbf{W}_j + A_{rj}\mathbf{W}_k + A_{rk}\mathbf{W}_j \\ \mathbf{W}_j A'_{rj} + \mathbf{W}_j A'_{rk} + \mathbf{W}_k A'_{rj} & -\rho(2\mathbf{W}_j + \mathbf{W}_k) \end{pmatrix} < \frac{1}{(N-1)^2} \begin{pmatrix} \mathbf{I} & 0 \\ 0 & 0 \end{pmatrix} \tag{8}$$
$$j = 1, ..., N \quad ; \quad k = 1, ..., N \quad ; \quad k \neq j$$

$$\begin{pmatrix} A_{rj}\mathbf{W}_k + A_{rk}\mathbf{W}_j + \mathbf{W}_j A'_{rk} \\ +\mathbf{W}_k A'_{rj} + A_{rj}\mathbf{W}_l + A_{rl}\mathbf{W}_j & A_{rk}\mathbf{W}_j + A_{rj}\mathbf{W}_k + A_{rl}\mathbf{W}_j \\ +\mathbf{W}_j A'_{rl} + \mathbf{W}_l A'_{rj} + A_{rl}\mathbf{W}_k & +A_{rj}\mathbf{W}_l + A_{rl}\mathbf{W}_k + A_{rk}\mathbf{W}_l \\ +A_{rk}\mathbf{W}_l + \mathbf{W}_l A'_{rk} + \mathbf{W}_k A'_{rl} & \\ \mathbf{W}_j A'_{rk} + \mathbf{W}_k A'_{rj} + \mathbf{W}_j A'_{rl} & -2\rho(\mathbf{W}_j + \mathbf{W}_k + \mathbf{W}_l) \\ +\mathbf{W}_l A'_{rj} + \mathbf{W}_k A'_{rl} + \mathbf{W}_l A'_{rk} & \end{pmatrix} < \frac{6}{(N-1)^2} \begin{pmatrix} \mathbf{I} & 0 \\ 0 & 0 \end{pmatrix} \tag{9}$$
$$j = 1, ..., N-2 \quad ; \quad k = j+1, ..., N-1 \quad ; \quad l = k+1, ..., N$$

$$\begin{pmatrix} sin\theta(A_j\mathbf{W}_j + \mathbf{W}_j A'_j) & cos\theta(A_j\mathbf{W}_j - \mathbf{W}_j A'_j) \\ cos\theta(-A_j\mathbf{W}_j + \mathbf{W}_j A'_j) & sin\theta(A_j\mathbf{W}_j + \mathbf{W}_j A'_j) \end{pmatrix} < -\mathbf{I} \quad ; \quad j = 1, ..., N \tag{11}$$

$$\begin{pmatrix} sin\theta(A_j\mathbf{W}_k + A_k\mathbf{W}_j + \mathbf{W}_j A'_k + \mathbf{W}_k A'_j) & cos\theta(A_j\mathbf{W}_k + A_k\mathbf{W}_j - \mathbf{W}_j A'_k - \mathbf{W}_k A'_j) \\ cos\theta(-A_j\mathbf{W}_k - A_k\mathbf{W}_j + \mathbf{W}_j A'_k + \mathbf{W}_k A'_j) & sin\theta(A_j\mathbf{W}_k + A_k\mathbf{W}_j + \mathbf{W}_j A'_k + \mathbf{W}_k A'_j) \end{pmatrix} < \frac{2}{N-1}\mathbf{I} \tag{12}$$
$$j = 1, ..., N-1 \quad ; \quad k = j, ..., N$$

the limits of α_i, which are found by checking the feasibility of the LMI conditions.

A problem particular to each system consists in finding the state matrix A and defining the perturbation matrices E in a way that it is possible to overcome the usually non linear relationships between the elements in the original matrix A and the parameters of the system. Perturbation matrices E_i must be defined carefully, since the same parameter can appear in several positions of the matrix.

Considering the developments shown in this section, we derive the following proposition.

Proposition 1. *For a given uncertain system* (4)*, if there exist positive definite matrices W_i, $\forall i = 1, \ldots, N$, such that the conditions* (7)*,* (8)*,* (9)*,* (11) *and* (12) *are satisfied, then the stability of matrix $A(p)$ and its pole-placement performance in the subset formed by the circle of radius ρ and the conic sector of angle θ is guaranteed.*

Proof 1. *Directly follows from the elements given in Sections II-A and II-B.*

III. Bisection Algorithm

Proposition 1 allows to verify if a range of parameters in matrix A satisfy the conditions mentioned above. However the objective of the paper is to find this range of parameters in matrix A. Finding how large the perturbation amount can be is the job of the bisection algorithm.

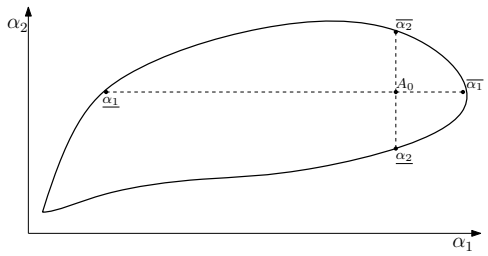
Given an initial feasible A_0 matrix and perturbation directions E_i, the bisection algorithm finds a feasible range of $\left[-\underline{\alpha_i}, \overline{\alpha_i}\right]$ $\forall i = 1, ..., m$.

An additional boolean vector ψ_i is used in order to define the desired "direction of movement" for each α_i. The vector is defined as $\psi_i := [\underline{\psi_i},\ \overline{\psi_i}]$, $\forall i = 1, ..., m$. As an example, for ψ_i = [0, 1], the search is restricted to positive values of α_i, and the corresponding vector $\left[-\underline{\alpha_i}, \overline{\alpha_i}\right]$ is $[0, \overline{\alpha_i}]$.

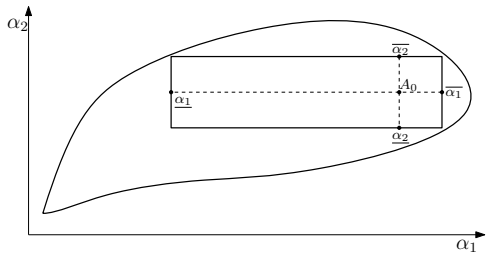
The directionality can be useful to only increase or to only decrease some particular parameter through α_i. As an example in the context of power electronics, the search for smaller parameters may translate to smaller components and therefore lower costs. Conversely, the search for larger parameters may be related to parasitic elements, in order to assess the influence of these undesired factors in the dynamic behaviour.

The search of the feasible domain is carried out in two steps. First, the maximum independent non-symmetric domain for each individual perturbation direction α_i is found. This step is noted as *independent* because only one perturbation direction is tackled at a time. It is *non-symmetric* since the distances between A_0 and the limits of the stability domain are not necessarily symmetric i.e. maximum and minimum α_i values $\left[-\underline{\alpha_i}, \overline{\alpha_i}\right]$ do not necessarily have the same magnitude. Figure 2a shows the idea of these domains for case in which $N = 2$. The domain is found by bisection, i.e.: testing the feasibility or the unfeasibility of the LMI conditions.

The resulting feasible domains are used as a scaling parameter to obtain the *simultaneous non-symmetric* domain. Using

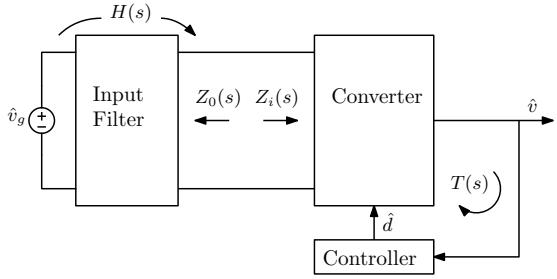


(a) Maximum independent non-symmetric domain example.

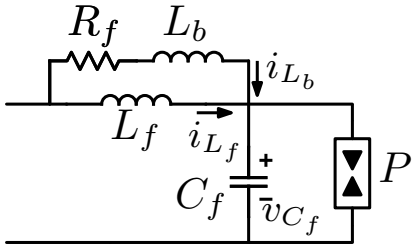


(b) Simultaneous non-symmetric domain example.

Fig. 2: Bisection algorithm domains examples



(a) Generic filter plus closed loop converter.



(b) LC filter with parallel R_f and L_b branch, loaded by a CPL.

Fig. 3: Generic and simplified models

the previous domain as a scaling parameter helps in dealing with the case when there is a big discrepancy in the possible range of movement between directions. *Simultaneous* means that all the E_i are used simultaneously to create the polytope to be tested by the LMIs. Again, this domain is found by bisection. Now the bisection only works on a single scalar value of α and the individual α_i are obtained as a product of α and the scaling parameter. Figure 2b shows an example of how the final polytope may look like.

The algorithm described above can be posed as follows:

Algorithm 1.

1. Define the problem description: matrices $A_0,\ E_i,\ \psi_i$ *and requirements* ρ_{max}, θ_{max}.

2. Obtain the independent non-symmetric domains.

$\forall\ i = 1, ..., m$

if $\underline{\psi_i} = true,\quad \max(\underline{\alpha_i})$ subject to LMIs (7), (8), (9), (11), (12).

else $\underline{\alpha_i} = 0$.

if $\overline{\psi_i} = true,\quad \max(\overline{\alpha_i})$ subject to LMIs (7), (8), (9), (11), (12).

else $\overline{\alpha_i} = 0$.

3. Obtain the simultaneous non-symmetric domain.

$\max(\alpha)$ subject to LMIs (7), (8), (9), (11), (12).

4. Each individual α_i *is obtained by* α *and the scaling parameter.*

This final domain is used to obtain the range of feasible parameters of the system. Any combination of the α_i obtained is feasible. This is also true for the parameters of the system if they are independent with respect to each other.

IV. Case Example: Robust Filter Design

The basis for what can be considered conventional filter design can be seen on [1]–[3]. These conventional methods rely on simplifications of the stability criteria that might lead to conservative results. Figure 3a shows a typical representation of the design problem, a power converter with a control loop and an input filter. In Figure 3b, the converter is simplified as a constant power load (CPL), and the filter is a third order case. The tools described above are used to perform input filter design for two different filters.

Reducing the size of the filter is one of the objectives, which is in accordance with decreasing the capacitance and the inductance of the elements in the filter. A feasible oversized filter can be used as a starting point for the search of the feasible domain of parameters. As illustrated in Figure 4 the feasible domain depends on the initial system A_0.

Therefore, a recursive approach that runs Algorithm 1 and finds new matrices $A_{d\text{new}}$ is proposed. This iterative process will be finished when the range of the filter parameters (P_i) obtained is lower than some given threshold Δ_{p_i}. A final bisection search will be performed, with the objective of finding the largest possible region of parameters satisfying the dynamical properties. This final search will start at the most favourable point found and go in the opposite direction in order to find a large parameter region. The algorithm used is defined on Algorithm 2.

Algorithm 2.

1. Set the specifications of the problem.

2. Check A_0 *feasibility with* $\alpha_i = 0$. *Otherwise an error is returned.*

3. Run Algorithm 1 to obtain the α_i.

4. Compute A_{dnew} *with an average value of* α_i.

5. If $(P_{iold} - P_{inew} > \Delta_i) \rightarrow A_0 = A_{dnew}$ *and go back to 3, else obtain* A_0 *with the maximum values of* α_i.

6. Run Algorithm 1 with $\psi_i = 1 - \psi_i$.

7. Compute the final parameter region A_0 *and* α_i.

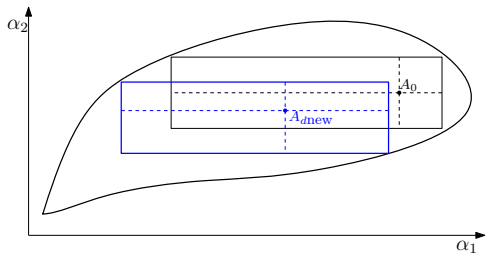


Fig. 4: Example of the domain obtained by the bisection algorithm with A_0 and the domain obtained with $A_{d\text{new}}$.

A. Case Example: LC Filter with parallel L-R damping branch

In this filter topology, shown in Figure 3b, the aim is to reduce the filter parameters while being able to obtain a parameter region from which a designer can extract viable filter parameters. The damping is given by a L-R branch in parallel with the filter inductance. The losses are low since the DC power does not involve the damping resistor. Such system has the following state space representation A matrix:

$$A = \begin{pmatrix} 0 & 0 & \frac{-1}{L_f} \\ 0 & \frac{-R_f}{L_b} & \frac{-1}{L_b} \\ \frac{1}{C_f} & \frac{1}{C_f} & \frac{\beta}{C_f} \end{pmatrix}, \text{with } x = \begin{pmatrix} i_{L_f} \\ i_{L_b} \\ v_{C_f} \end{pmatrix} \tag{14}$$

as the state vector, and $\beta = \frac{P}{V^2}$, P being the set power and V the voltage on the CPL. The following perturbation matrices will be defined:

$$E_1 = \begin{pmatrix} 0 & 0 & \frac{-1}{L_{f_{sc}}} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad E_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{C_{f_{sc}}} & \frac{1}{C_{f_{sc}}} & \frac{\beta}{C_{f_{sc}}} \end{pmatrix}$$
$$E_3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{-R_{f_0}}{L_{b_{sc}}} & \frac{-1}{L_{b_{sc}}} \\ 0 & 0 & 0 \end{pmatrix} \tag{15}$$

This means that the uncertain model will have $N = 2^3 = 8$ vertices. The suffix "$_{sc}$" is short for "scaling". All the parameters with this suffix are used in order to keep the perturbation values as close as possible to each other. The designed filter parameters can be obtained by

$$A_{des} = A_0 + \sum_{i=1}^{k} \alpha_i E_i \tag{16}$$

where A_{des} and A_0 have the same structure as A. The initial parameters of the filter in A_0 are noted with the suffix "$_0$", whereas the range of parameters that has been found is noted with the suffix "$_d$".

Param.	Initial Values	Parameter Ranges	Selected Values
L_f/µH	500	7.82 - 246e6	10
C_f/µF	100	21.3 - 83.4	33
L_b/µH	150	1.96 - 2.91	2.2

TABLE I: LC with parallel RL branch numerical example

In the case at hand, each of the parameters in the filter only depends on one particular α_i and E_i (they are independent from each other), thus they can be found as follows:

$$L_{f_d} = (-A_{dj}(1,3))^{-1} = \left(\frac{1}{L_{f_0}} + \alpha_1 \frac{1}{L_{f_{sc}}}\right)^{-1} \tag{17}$$

$$C_{f_d} = (A_{dj}(3,1))^{-1} = \left(\frac{1}{C_{f_0}} + \alpha_2 \frac{1}{C_{f_{sc}}}\right)^{-1} \tag{18}$$

$$L_{b_d} = (-A_{dj}(2,3))^{-1} = \left(\frac{1}{L_{b_0}} + \alpha_3 \frac{1}{L_{b_{sc}}}\right)^{-1} \tag{19}$$

For this particular filter, only one search is performed, implementing the three perturbation directions simultaneously. Note that, since α_1, α_2 and α_3 are vectors of dimension 2, a range for each parameter L_f, C_f and L_b is found. The damping resistance R_d is obtained as the optimum value for damping the filter according to [7]. The search has been programmed such that after each run of the bisection algorithm a new R_d is obtained which will be used to update the A_0 matrix.

For the numerical example, performance conditions have been set to: attenuation of at least $-0\,\text{dB}$ at the switching frequency of $f_{sw} = 250\,\text{kHz}$ and a damping ratio of at least $\zeta = 0.3$.

The attenuation is enforced with the use of ρ, since the poles of the system are in agreement with the magnitude asymptotes of the input-to-output current transfer function. Limiting the frequency of the poles sets a minimum distance between them and the switching frequency f_{sw}, at which a specific attenuation is desired. For a given attenuation $atten$, and a slope of the high frequency asymptote of the filter dB/Oct, ρ is found as follows:

$$\rho = \frac{2\pi f_{sw}}{2^p} \; ; \; p = \frac{atten}{\text{dB}/Oct} \tag{20}$$

Then, the minimum damping ratio is enforced through the conic region of angle θ. For a given minimum damping ratio ζ, θ is found as follows:

$$\theta = cos^{-1}(\zeta). \tag{21}$$

Table I shows numerical values tested for this type of filter, including the initial values from which Algorithm 2 has started, the resulting range of filter parameters and finally a possible filter option. Note that the initial system is well oversized. The methods described above are able to reduce the overall size of the components and provide the designer with a range of parameters. From that range, the design engineer can choose the most suitable values according to parameters other than stability or dynamical performance. It can be seen that all of the parameters have a range that allows to easily find commercially available components. The huge upper range of

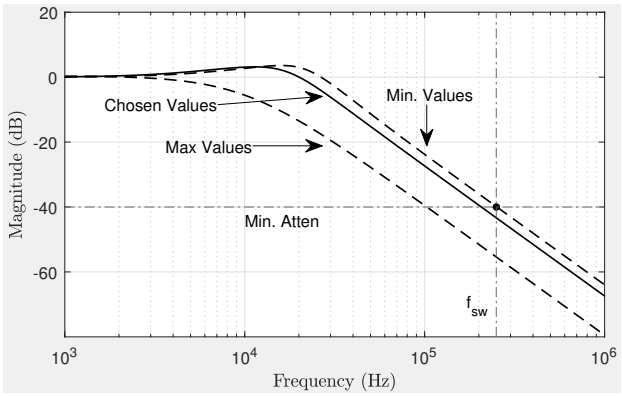


Fig. 5: Input current - to - output current $\left(\frac{i_o}{i_{in}}\right)$ frequency response of the LCRL filter. The solid line is for the chosen values and the dashed lines represent the extreme vertices of the polytope, i.e. the frequency response with the maximum and the minimum values of the parameters.

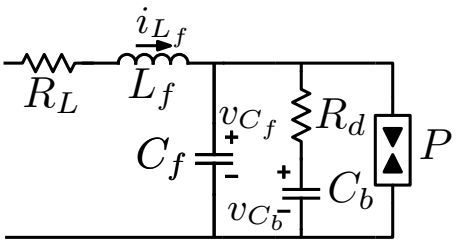


Fig. 6: LC filter with parallel RC branch.

L_f indicates that in practice there is not limit on how large that inductor can be.

Figure 5 shows the frequency response of the final filter (as well as the maximum and minimum values). Attenuation at f_{sw} is higher than the specifications. It is also possible to check that the damping is higher than the specifications.

B. Case Example: LC Filter with RC Parallel Damping

The schematic for this type of filter can be seen on Figure 6. For this case the state matrix of the system is

$$A = \begin{pmatrix} \frac{-R_L}{L_f} & \frac{-1}{L_f} & 0 \\ \frac{1}{C_f} & \frac{\beta}{C_f} - \frac{1}{R_d C_f} & \frac{1}{R_d C_f} \\ 0 & \frac{1}{R_d C_b} & \frac{-1}{R_d C_b} \end{pmatrix}, \text{with } x = \begin{pmatrix} i_{L_f}(t) \\ v_{C_f}(t) \\ v_{C_b}(t) \end{pmatrix} \tag{22}$$

as the state vector, and $\beta = \frac{P}{V^2}$. The perturbation matrices are defined as:

$$\begin{aligned} E_1 &= \begin{pmatrix} \frac{-R_L}{L_{f_{sc}}} & \frac{-1}{L_{f_{sc}}} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \\ E_2 &= \begin{pmatrix} 0 & 0 & 0 \\ \frac{1}{C_{f_{sc}}} & \frac{\beta}{C_{f_{sc}}} - \frac{1}{R_d C_{f_{sc}}} & \frac{1}{R_d C_{f_{sc}}} \\ 0 & 0 & 0 \end{pmatrix}, \\ E_3 &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{-1}{R_{d_{sc}} C_f} & \frac{1}{R_{d_{sc}} C_f} \\ 0 & 0 & 0 \end{pmatrix}, \\ E_4 &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{R_{d_{sc}} C_{b_{sc}}} & \frac{-1}{R_{d_{sc}} C_{b_{sc}}} \end{pmatrix}. \end{aligned} \tag{23}$$

In order to show the versatility of the method, we employ a different approach in this case. Algorithm 2 is employed twice, for two different sets of parameters of the filter. The first time, the algorithm finds a range for the values of L_f and C_f (E_1 and E_2). The second time, the algorithm looks for the maximum admissible range of values of R_d and C_b (E_3 and E_4). Since only two perturbation matrices are applied at the same time the resulting uncertain model will have $N = 2^2 = 4$ vertices. The range of parameters is obtained by:

$$L_{f_d} = -A_d(1,2)^{-1} = -\left(\frac{1}{L_0} + \alpha_1 \frac{1}{L_{sc}}\right)^{-1} \tag{24}$$

$$C_{f_d} = A_d(2,1)^{-1} = \left(\frac{1}{C_0} + \alpha_2 \frac{1}{C_{sc}}\right)^{-1} \tag{25}$$

$$R_{d_d} = (C_f A_d(2,3))^{-1} = \left(\frac{1}{R_{d_0} 1} + \alpha_3 \frac{1}{R_{d_{sc}}}\right)^{-1} \tag{26}$$

$$C_{b_d} = (R_{d_d} A_d(3,2))^{-1} = \left(\frac{R_{d_d}}{R_{d_0} C_{b_0}} + \alpha_4 \frac{R_{d_d}}{R_{d_{sc}} C_{b_{sc}}}\right)^{-1} \tag{27}$$

Tables II and III show the results of the example design. The performance conditions are an attenuation of $-80\,\text{dB}$ at the frequency of $250\,\text{kHz}$ and a damping ratio of $\zeta = 0.3$.

As it can be seen in the table, the initial values of the filter are oversized to $L = 470\,\mu\text{H}$, $C = 800\,\mu\text{F}$, $R_d = 1\,\Omega$, and $C_b = 2000\,\mu\text{F}$. The first search returns a range of values from which the designer can choose: $L \in 73.2 - 1890\,\mu\text{H}$ and $C \in 59.3 - 162\,\mu\text{F}$. Once these components are chosen to be $L = 220\,\mu\text{H}$ and $C = 100\,\mu\text{F}$, the second search returns another range of values for R_d and C_b: $R_d \in 0.67 - 1.5\,\Omega$ and $C_b \in 458 - 410e9\,\mu\text{F}$. In practice, such a large upper bound means that there is no limit in choosing a large capacitor Cb. Notice that since the parameters R_d and C_b depend on each other (27) even though the alpha region is an orthogonal plolytope the filter parameter region is not. In order to show a range of parameters that are all compatible with each other, the region previously mentioned has been conservatively constrained to be orthogonal. It could be the case that this simplification limits the values too much, in that case it would be necessary to look at the non-orthogonal region.

Figure 7 shows the magnitude frequency response of the possible and chosen filter values. It can be seen that the

Param.	Initial values	**Result. Range**	First Selection
$L_f/\mu\text{H}$	470	**73.2 - 1890**	220
$C_f/\mu\text{F}$	800	**59.3 - 162**	100
R_d/Ω	1	1	1
$C_b/\mu\text{F}$	2000	2000	2000

TABLE II: Filter values after L_f and C_f iteration

Param.	First Selection	**Result. Range**	Final Selection
$L_f/\mu\text{H}$	220	220	220
$C_f/\mu\text{F}$	100	100	100
R_d/Ω	1	**0.67 - 1.5**	0.7
$C_b/\mu\text{F}$	2000	**458 - 410e9**	940($2x470$)

TABLE III: Filter values after R_d and C_b iteration

attenuation at $f_{sw} = 250\,\text{kHz}$ is better than the specification of $-80\,\text{dB}$ set by the design constraints. The damping constraints are also met.

C. Influence of Initial Conditions

One of the possible problems of the proposed approach is its sensitivity to different initial conditions in A_0. Although there is no guarantee that the proposed method will achieve good results in all possible cases, the results have been found to present reasonable robustness to changes in A_0. In order to illustrate this point, a few tests are shown. The tests compare the results in the previous section with results obtained with different A_0.

The example from Section IV-A has been recalculated with smaller and larger initial parameters. The results are given on Table IV. First, it can be seen how the lower bound for L_f, C_f and L_b is relatively similar. Also, for different initial conditions the final ranges of values present differences, but they still allow the designer to choose similar or even smaller components.

The case in Section IV-B has been tested against initial conditions with values 40 % higher or lower than those in the original A_0. The resulting ranges can be seen in Table V. The resulting region still allows the designer to choose the same final values. In addition, it can be seen how the case that allows the lowest bound in R_d is also the case that exhibits a higher lower bound in C_b. This illustrates the trade-off between the different ranges in the components.

D. Numerical Complexity

The method proposed in the paper employs LMIs that are efficiently solved by interior-point methods. The complexity of the problem can be computed in terms of the number of parameters to be evaluated m and the size of the Lyapunov matrices, which is linked to the number of states in A, i.e. n. Additionally, since the algorithms rely on a feasibility test, the precision tolerance of the bisections also plays a role.

In the examples demonstrated above, conventional computers at the time of write-up of the paper, using MATLAB with YALMIP [8] and MOSEK [9] can solve Algorithm 1 with running times in the order of seconds up to one or two tens of seconds. Then, Algorithm 2 calls Algorithm 1 around ten times per design cycle which gives a rough estimate of how long the whole procedure can take.

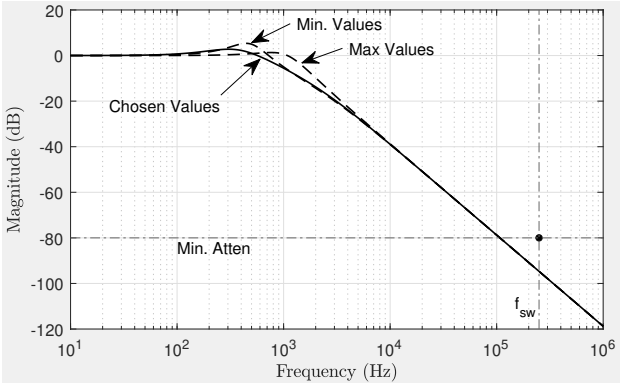


Fig. 7: Input current - to - output current $\left(\frac{i_o}{i_{in}}\right)$ frequency response of the filter.

Original initial values		
Param.	Initial values	Parameter ranges
$L_f/\mu\text{H}$	500	$7.82 - 246e6$
$C_f/\mu\text{F}$	100	$21.3 - 83.4$
$L_b/\mu\text{H}$	150	$1.96 - 2.91$
Halved initial values		
Param.	Initial values	Parameter ranges
$L_f/\mu\text{H}$	250	$5.78 - 17.92$
$C_f/\mu\text{F}$	50	$17.9 - 24.2$
$L_b/\mu\text{H}$	75	$1.93 - 2.76$
Doubled initial values		
Param.	Initial values	Parameter ranges
$L_f/\mu\text{H}$	1000	$6.88 - 217$
$C_f/\mu\text{F}$	200	$27.44 - 110$
$L_b/\mu\text{H}$	300	$1.53 - 2.03$

TABLE IV: Comparison of different initial points for the LC filter with parallel RL branch

V. Conclusions

This paper presented a numerical approach to physical parameter selection by means of uncertain models. LMIs have been used to implement performance conditions which force a minimum attenuation and damping of the system. A modified bisection algorithm has been used in order to find a region of parameters compatible with the performance conditions. It has been shown that it is possible to start with large parameters and iterate to find parameter regions with smaller overall parameters until certain tolerance conditions are met.

It is important to keep in mind that this approach may lead to conservative results and thus it does not guarantee the most optimised values. Nevertheless it gives the designer a tool to quickly estimate a feasible set of parameters.

A couple of design examples have been shown. Two different filter topologies have been used in order to implement the numerical methods described. Any filter value defined inside the found parameter region can be chosen according to other design criteria such as cost, size or other arbitrary criteria.

Original initial values		
Param.	Initial values	**Resulting range**
L_f/µH	220	220
C_f/µF	100	100
R_d/Ω	1	**0.67 - 1.5**
C_b/µF	2000	**458 - 410e9**
Initial values reduced by 40 %		
Param.	Initial values	**Resulting range**
L_f/µH	220	220
C_f/µF	100	100
R_d/Ω	0.6	**0.59 - 2.02**
C_b/µF	1200	**907 - 258e9**
Initial values increased by 40 %		
Param.	Initial values	**Resulting range**
L_f/µH	220	220
C_f/µF	100	100
R_d/Ω	2	**0.67 - 1.4**
C_b/µF	4000	**432 - 436e9**

TABLE V: Filter values after L_f and C_f iteration, with double initial values

REFERENCES

[1] R. W. Erickson and D. Maksimovic, *Fundamentals of Power Electronics 2nd Edition*, 2008, no. February.

[2] R. D. Middlebrook, "Input Filter Considerations in Design and Application of Switching Regulators," *IEEE Industry Applications Society Annual Meeting*, pp. 366 – 382, 1976.

[3] ——, "Design Techniques for Preventing Input Filter Oscillations in Switched-Mode Regulators," *Proceedings of Powercon 5*, pp. A3.1 – A3.16, 1978.

[4] D. C. Ramos and P. L. Peres, "An LMI approach to compute robust stability domains for uncertain linear systems," *Proceedings of the American Control Conference*, vol. 5, no. 2, pp. 4073–4078, 2001.

[5] V. J. Leite, V. F. Montagner, and P. L. Peres, "Robust pole location by parameter dependent state feedback control," *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, no. December, pp. 1864–1869, 2002.

[6] M. Chilali and P. Gahinet, "H_∞ Design with Pole Place Constraints: An LMI Approach," *Ieee Transactions on Automatic Control*, vol. 41, no. 3, pp. 358–367, 1996.

[7] R. W. Erickson, "Optimal single resistor damping of input filters," *Conference Proceedings - IEEE Applied Power Electronics Conference and Exposition - APEC*, vol. 2, pp. 1073–1079, 1999.

[8] J. Löfberg, "YALMIP," 2022. [Online]. Available: https://yalmip.github.io/download/

[9] MOSEK ApS, "The MOSEK optimization toolbox for MATLAB. Version 9.3.18," 2022. [Online]. Available: https://www.mosek.com/downloads/