Simultaneous automated design of structured QFT controller and prefilter using nonlinear programming

H. Purohit[3] A. Goldsztejn[1] C. Jermann[2]
L. Granvilliers[2] F. Goualard[2] P. S. V. Nataraj[3]
B. Patil[4]

[1] CNRS, IRCCyN, France
[2] University of Nantes, France
[3] IIT Bombay, India
[4] CARES, Singapore

June 8, 2016
preprint

Abstract

This paper describes a nonlinear programming based robust design methodology for controllers and prefilters of a predefined structure for the linear time invariant systems involved in the quantitative feedback theory (QFT). This controller and prefilter synthesis problem is formulated as a single optimization problem with a given performance optimization objective and constraints enforcing stability and various specifications usually enforced in the QFT. The focus is set on providing constraints expression that can be used in standard nonlinear programming solvers. The nonlinear solver then computes in a single step controller and prefilter design parameters that satisfy the prescribed constraints and maximizes the performance optimization objective. The effectiveness of the proposed approach is demonstrated through a variety of difficult design cases like resonant plants, open loop unstable plants and plants with variation in the time delay.

Keywords Quantitative Feedback Theory, Controller and prefilter synthesis, nonlinear programming

1 Introduction

Most of the real world systems are uncertain because of external disturbances and measurements noise. Due to this uncertainty, there are always mismatches between

[2]E-mail: harsh.purohit@sc.iitb.ac.in

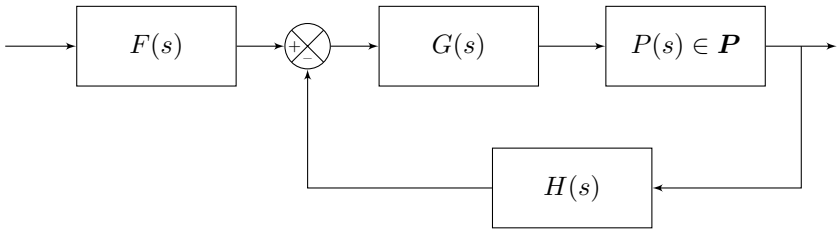


Figure 1: Two degrees-of-freedom structure in QFT.

mathematical models and actual real systems. Robust control design techniques are well-suited for this kind of uncertain systems. Because of its usefulness, the automation of robust controllers and prefilters design is of key concern in the control community [11]. Quantitative feedback theory (QFT), introduced by Isaac Horowitz, is one such frequency domain technique which uses Nichols charts in order to achieve a desired robust design over a specified region of uncertainty in the plant parameters [22]. It uses a two-degrees-of-freedom structure as shown in Figure 1. The main objective of QFT is to find the controller and prefilter transfer functions which guarantee the system performance specifications over the range of plant uncertainties and over the range of design frequencies.

QFT design technique uses phase information in the design process and it is highly transparent to see design trade-offs at each design frequency. QFT consists of design steps like bound generation and controller and prefilter synthesis using loop-shaping methods. The automation of loop-shaping is of great interest because even for a skillful person it becomes very tedious and time consuming for complex systems.

Automatic loop-shaping technique can be classified into two main categories:

(i) Based on optimization algorithm such as convex, non-convex optimization (see, e.g., [36, 10, 33]).

(ii) Based on constraint satisfaction approach which gives dense set of controllers from which the desired solution can be chosen (see, e.g., [38, 28, 31]).

Initially, Horowitz and Gera [15] have described a semi-automatic loop-shaping process which contains an iterative approach and adds an element to the transfer function to force the open loop transfer function to satisfy the Nichols chart specification. Thompson and Nwokah [34], proposed a constrained, nonlinear programming approach that starts from an assumed initial QFT controller. Zolotas and Halikias [40] have proposed an automatic loop-shaping method for Proportional Integral Derivative (PID) controllers based on searching over a dense set of controllers. This technique is efficient for two parameters controller and a small number of bounds. Yaniv and Naurka [38] propose a method based on non-iterative optimization for two parameters controllers.

Nataraj et. al [30] reformulated the problem as the parametric optimization of a fixed structure controller for single-input single-output (SISO) systems. A numerical constraint satisfaction problem based approach for automated synthesis of fixed structure integer-order QFT controller for SISO system has been proposed in [27]. Cervera et al. [5] proposed optimization based algorithm for the design of robust controller using CRONE (Commande Robuste d'Ordre Non Entier) structure. Recently, several researchers have attempted to automate the prefilter synthesis step for SISO

system [28]. Nataraj and Tharewal [29] formulated the problem as the parametric optimization of a fixed structure integer order prefilter using global optimization.

In all approaches from the QFT literature, the controller and prefilter synthesis problems are addressed sequentially in two steps: (i) The controller synthesis problem; (ii) The prefilter synthesis problem. The second step problem may be infeasible, i.e. there may be no prefilter corresponding to the controller computed in the first step. Backtracking on the first step to choose another controller is not only tedious and time consuming, but it also requires manual intervention of the designer, making the whole process partially automated only. Motivated by these limitations, we propose a novel approach to design controller and prefilter simultaneously, in a single step, as a constrained nonlinear robust optimization problem. The proposed approach should allow a fully automated (simultaneous) synthesis of a robust controller and an associated prefilter, which optimizes a given performance objective.

We also propose solutions to overcome two other obstacles for a complete automation of the process: First, the stability of the system is not explicitly encoded in the QFT constraints, stability being usually enforced manually by the designer during the classical loop-shaping process; Second, the usual QFT bound constraints are not expressions that can be handled by numerical optimizers.

The paper is organized as follows. In Section 2, robust control problem is discussed and existing solving strategies are described. Section 3 present the new 1-step approach we propose to solve the problem, and discusses its merits. Results and simulation are demonstrated in Section 4. Conclusion of the work are drawn in Section 5.

Notations. Intervals and interval vectors (cartesian product of intervals), also called boxes, are denoted by boldface symbols.

2 QFT formulation of the robust control problem

In this section, we recall the definition of the QFT problem, consisting of designing a controller and a prefilter for an uncertain linear time invariant system coming from the linearization of a nonlinear system. We also recall the classical resolution of this design problem.

2.1 The robust control problem

In any engineering system, often called *plant*, it is of fundamental importance that control systems be designed so that stability is preserved in the face of various classes of uncertainties. An uncertain linear time-invariant plant is given by $P_\lambda(s)$, where $\lambda = (\lambda_1, \ldots, \lambda_p)$ is a vector of uncertain plant parameters, real scalars whose values vary over a parameter box $\boldsymbol{\lambda}$. This gives rise to a parametric uncertain plant family

$$\boldsymbol{P} = \{P_\lambda(s) : \lambda \in \boldsymbol{\lambda}\}\,. \tag{1}$$

Consider the two-degrees-of-freedom feedback system configuration as shown in Figure 1 where $G_x(s)$ and $F_y(s)$ are the controller and prefilter transfer functions, respectively parameterized by a vector $x = (x_1, \ldots, x_n)$ and a vector $y = (y_1, \ldots, y_m)$ of design parameters whose definitions depend on the chosen structures. For the sake of

clarity sensor dynamics H have been considered to be unitary. The pure error feedback open-loop and closed-loop transfer functions are defined without the prefilter:

$$L_{x,\lambda}(s) = G_x(s)P_\lambda(s), \tag{2}$$

$$T_{x,\lambda}(s) = \frac{L_{x,\lambda}(s)}{1 + L_{x,\lambda}(s)}. \tag{3}$$

The QFT approach to the robust control problem is to compute design parameters values $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ that satisfy some constraints $g_i(x, y) \leq 0$ related to stability and performance, and that optimize some given performance objective function $c(x, y)$. Therefore, it actually has the form of a nonlinear program

$$\begin{array}{ll} \min & c(x,y) \\ s.t. & g_i(x,y) \leq 0 \\ & x \in \boldsymbol{x} \\ & y \in \boldsymbol{y} \end{array}, \tag{4}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are initial domains for the design parameters. In practice, we may not be able to find the minimizer of (4), but feasible values for x and y with good criterion evaluation could be sufficient. The constraints are expressed in terms of frequency response of the system[1] and include:

- The robust stability margin (RSM) constraint: For a given margin ω_s,

$$\forall \omega \in [0, \infty],\ \forall \lambda \in \boldsymbol{\lambda},\ |T_{x,\lambda}(j\omega)| \leq \omega_s. \tag{5}$$

- The robust performance tracking (RPT) constraints: For given tracking bounds $A(\omega) \leq B(\omega)$,

$$\forall \omega \in \boldsymbol{\omega},\ \forall \lambda \in \boldsymbol{\lambda},\ A(\omega) \leq |F_y(j\omega)|\,|T_{x,\lambda}(j\omega)|, \tag{6}$$

$$\forall \omega \in \boldsymbol{\omega},\ \forall \lambda \in \boldsymbol{\lambda},\ |F_y(j\omega)|\,|T_{x,\lambda}(j\omega)| \leq B(\omega), \tag{7}$$

where a bounded design frequency domain $\boldsymbol{\omega} = [0, \omega_h]$ is usually provided, which depends on the system.

An alternative performance tracking constraint was shown to fit the QFT framework in [12], which can also be included in the framework proposed in the present paper. In the same manner, other constraints like input/output disturbance rejection, noise rejection, control effort and model matching [7] can also be included in the design process. Though we don't address them explicitly in this paper, these additional constraints fit the approach we propose.

The stability of the closed-loop system is usually enforced for an arbitrary nominal plant (in the uncertain plant family) using the Nyquist stability criterion, and maintained during the controller[2] loop shaping process. Finding a prefilter and a controller that satisfy these constraints and optimize the objective function is referred to as *the QFT problem* in the rest of this paper.

All these constraints are actually universally quantified real constraints on the design parameters x and y, and have therefore the form $g(x, y) \leq 0$. E.g., (5)–(7) are

[1] Since systems of interest are stable, the transfer function Laplace transform converges for $s = jw$.

[2] Only stable prefilter structures are considered, hence the prefilter is not involved in the stability enforcement.

respectively equivalent to $g_{\mathrm{RSM}}(x) \leq 0$, and $g_{\mathrm{RPTL}}(x,y) \leq 0$ and $g_{\mathrm{RPTU}}(x,y) \leq 0$, with

$$g_{\mathrm{RSM}}(x) = \max_{\omega \in \boldsymbol{\omega}, \lambda \in \boldsymbol{\lambda}} \; |T_{x,\lambda}(j\omega)| - \omega_s \tag{8}$$

$$g_{\mathrm{RPTL}}(x,y) = \max_{\omega \in \boldsymbol{\omega}, \lambda \in \boldsymbol{\lambda}} \; A(\omega) - |F_y(j\omega)|\,|T_{x,\lambda}(j\omega)| \tag{9}$$

$$g_{\mathrm{RPTU}}(x,y) = \max_{\omega \in \boldsymbol{\omega}, \lambda \in \boldsymbol{\lambda}} \; |F_y(j\omega)|\,|T_{x,\lambda}(j\omega)| - B(\omega). \tag{10}$$

Such a non-convex nonlinear problem with universally quantified constraints having non-concave parametric dependences is difficult to solve. In fact, to the best of our knowledge, there currently exist no method for solving such problems. In the QFT framework, it is traditionally approximated by first splitting the problem into two subproblems involving respectively only x and only y, and eliminating the quantifiers by sampling the uncertainties. Both processes are described in the following two subsections.

2.2 Two-steps resolution

The two-steps resolution of the problem consists in first finding controller design parameters which satisfy the robust stability margin constraint, and second finding prefilter design parameters which satisfy the robust performance tracking constraint together with the controller design chosen in the first step. In such decoupled problem, the second step may have no solution corresponding to the controller parameters computed in the first step, i.e. the second step problem may be infeasible, which entails backtracking on the first step to choose another controller setting. In order to increase the chances to have a feasible second step problem, a necessary condition for the robust performance tracking constraints is usually used in the first step. Considering two arbitrary uncertainties λ_1 and λ_2, the robust performance tracking constraints entail the following new constraint, which we call the *robust performance tracking ratio* (RPTR) constraint in the rest of the paper:

$$\forall \omega \in \boldsymbol{\omega},\ \forall \lambda_1 \in \boldsymbol{\lambda},\ \forall \lambda_2 \in \boldsymbol{\lambda},\ \frac{|T_{x,\lambda_1}(j\omega)|}{|T_{x,\lambda_2}(j\omega)|} \leq \frac{B(\omega)}{A(\omega)}. \tag{11}$$

This is again a universally quantified real constraint acting on the controller design parameters x only, whose equivalent expression is $g_{\mathrm{RPTR}}(x) \leq 0$ with

$$g_{\mathrm{RPTR}}(x) \;=\; \max_{\omega \in \boldsymbol{\omega}, \lambda_1 \in \boldsymbol{\lambda}, \lambda_2 \in \boldsymbol{\lambda}} \; \frac{|T_{x,\lambda_1}(j\omega)|}{|T_{x,\lambda_2}(j\omega)|} - \frac{B(\omega)}{A(\omega)}. \tag{12}$$

Now formally, the two-steps approach consists in solving first

$$\begin{array}{rll} x_* = \mathrm{argmin} & c_1(x) & \\ s.t. & g_{\mathrm{RSM}}(x) \leq 0 \wedge g_{\mathrm{RPTR}}(x) \leq 0 & , \\ & x \in \boldsymbol{x} & \end{array} \tag{13}$$

and then

$$\begin{array}{rll} y_* = \mathrm{argmin} & c_2(y) & \\ s.t. & g_{\mathrm{RPTL}}(x_*,y) \leq 0 \wedge g_{\mathrm{RPTU}}(x_*,y) \leq 0 & , \\ & y \in \boldsymbol{y} & \end{array} \tag{14}$$

where the criteria $c_1(x)$ and $c_2(y)$ must be defined to correspond to the global objective criterion $c(x,y)$ in (4).

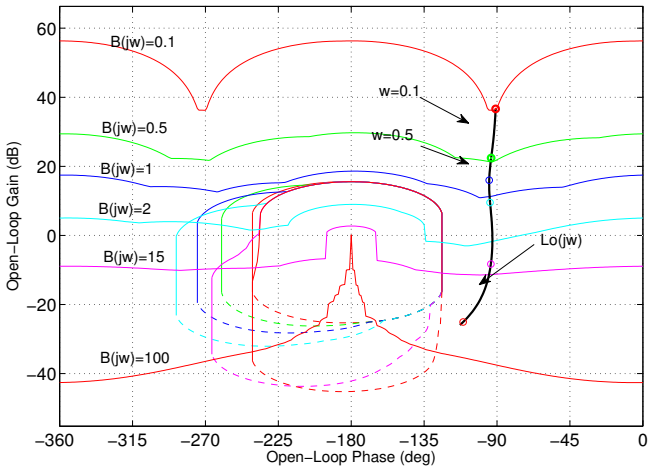


Figure 2: Nichols chart with bound constraints for Example 1

Again in practice, x_* and y_* may not be minimizers, but instead feasible vectors with good criteria evaluations. Note that since $g_{\text{RPTR}}(x) \leq 0$ is only necessary for $\exists y, g_{\text{RPTL}}(x,y) \wedge g_{\text{RPTU}}(x,y)$, Problem (14) may be infeasible while the original problem (4) could actually be feasible (this issue of the two-steps approach is well known, see e.g., [1]). This is illustrated by the following example.

Example 1 *Consider the standard QFT problem DC-motor [3, 28] defined by*

$$P(s) = \frac{k_P\, a}{s(s+a)}, \tag{15}$$

with uncertainty domains $a \in [1,10]$ *and* $k_P \in [1,10]$*, and frequency samples* $\boldsymbol{\omega} = \{0.1, 0.5, 1, 2, 15, 100\}$*. Furthermore,*

$$G(s) = k_G \frac{\frac{s}{z_1}+1}{(\frac{s}{p_1}+1)(\frac{s}{p_2}+1)} \tag{16}$$

$$F(s) = \frac{1}{(\frac{s}{q_1}+1)(\frac{s}{q_2}+1)}, \tag{17}$$

with design domains $k_G \in [0.1, 100]$*,* $z_1 \in [0.1, 50]$*,* $p_i \in [0.1, 1000]$*,* $q_i \in [0, 100]$*. The solution reported in in [28] is* $k_G = 10.95$*,* $z_1 = 2.1$*,* $p_1 = 950$*,* $p_2 = 982$*,* $q_1 = 3.197$*,* $q_2 = 8.61$*. Optimizing* $c_1(x) := k_G$*, the controller of this solution can be improved to* $k_G = 8.00$*,* $z_1 = 1.90$*,* $p_1 = 555.65$*,* $p_2 = 538.81$ *during the first step phase, which is proved to satisfy the RSM and RPTR constraints, e.g. using complete global constraint solver like IBEX [6, 35, 2] or Realpaver [16][3] (see Figure 2, which displays the bound constraints for this controller and shows they are indeed satisfied). However, it can be proved, using the same constraint solvers, that Problem* (14) *has no prefilter solution satisfying the RPT constraints for this controller.*

[3]Once the design variables x are fixed to some values x_*, verifying a universally quantified constraint $\forall \omega \in \boldsymbol{\omega}, \forall \lambda \in \boldsymbol{\lambda},\ g(\omega, \lambda, x_*) \leq 0$ is equivalent to checking that $g(\omega, \lambda, x_*) > 0$ has no solution inside $\boldsymbol{\omega}$ and $\boldsymbol{\lambda}$, which can be verified using standard numerical constraint solvers.

2.3 QFT bounds

The constraints $g_{\text{RSM}}(x) \leq 0$, $g_{\text{RPTL}}(x,y) \leq 0$, $g_{\text{RPTU}}(x,y) \leq 0$ and $g_{\text{RPTR}}(x) \leq 0$ can hardly be used directly because of the universal quantifiers (or, equivalently, maximum operators) they involve. The classical approach to overcome this difficulty is to sample the universally quantified domains. First, frequency design samples $\widetilde{\boldsymbol{\omega}} = \{\omega_1, \ldots, \omega_q\} \subseteq \boldsymbol{\omega}$ are chosen. Then for each frequency sample, plant uncertainties are sampled leading to bound constraints.

QFT bounds express the robust stability margin constraints (5) and the robust performance tracking ratio constraints (11) with respect to the magnitude g and phase ϕ of the controller. Other constraints like input/output disturbance rejection, noise rejection, control effort and model matching can also give rise to such bound constraints. The algorithm proposed in [7] relies on the quadratic nature of the original constraints for fixed controller phase and uncertainties λ. E.g., the robust stability margin constraint (5) has the following form:

$$g^2 p_\lambda^2 \left(1 - \frac{1}{\omega_s^2}\right) + 2g p_\lambda \cos(\phi + \theta_\lambda) + 1 \geq 0, \tag{18}$$

where p_λ (resp. θ_λ) is the plant magnitude (resp. phase) for uncertainty λ. This constraint is quadratic with respect to g for fixed ϕ, p_λ and θ_λ. Bound constraints are sometimes equivalently expressed with respect to a nominal open loop magnitude $|L_0| = g p_{\lambda_0}$ and phase $\angle L_0 = \phi + \theta_{\lambda_0}$, for an arbitrary nominal parameter $\lambda_0 \in \boldsymbol{\lambda}$, in which case (18) is replaced by

$$|L_0|^2 \left(\frac{p_\lambda}{p_{\lambda_0}}\right)^2 \left(1 - \frac{1}{\omega_s^2}\right) + 2|L_0| \frac{p_\lambda}{p_{\lambda_0}} \cos(\angle L_0 + \theta_\lambda - \theta_{\lambda_0}) + 1 \geq 0, \tag{19}$$

which is also quadratic with respect to $|L_0|$ for fixed $\angle L_0$, p_λ and θ_λ. Similar constraints with quadratic form in g or $|L_0|$ can also be derived for the other constraints.

The algorithm in [7] computes explicitly the bounds for the magnitude of the controller for fixed phase and uncertainties for each sample of plant uncertainties and each sample of controller (or open-loop) phase, giving rise to a discretized approximation of the boundary of the original constraint in the controller (or open-loop) magnitude and phase plane. Improving the efficiency and accuracy of this algorithm for the bound generation has been the topic of several papers (see, e.g., [26, 4, 37] for efficiency improvements, and [23, 14] for an efficient analytical derivation of the bound constraints).

Note here that this process involves discretizing both the plant uncertainties and the controller phase. Furthermore, the sampled description of the constraint under the form of a list of points describing the boundary of the constraint, although very useful for a manual graphical solving process, is by nature difficult to use within automatic numerical algorithms, which usually require constraints of the form $g(x) \leq 0$ where g can be evaluated and differentiated. Nevertheless, QFT bounds have been tackled directly using two kind of numerical algorithms: First, QFT bounds were successfully used to contract a box by listing the points it contains in [33], within a nonlinear branch and bound algorithm. However, this process limits the convergence of the algorithm since such a discretized description cannot be differentiated, while efficient optimization algorithms (local or global) require differentiating at least once the constraints. Second, QFT bounds have been used directly using population based optimization algorithms: [10, 9] use a simple constraint which is either 0 or 1 depending

on the bound satisfaction, while [25] uses a more elaborated constraint which compute a distance to the bound satisfaction.

3 A fully automated one-step nonlinear programming approach

As emphasized in the previous section, splitting the resolution of the QFT problem into two steps raises one important issue: The controller solution computed in the first step may turn out to have no corresponding prefilter solution in the second step. This is even more likely to happen when efficient numerical solvers are used, since they will attempt to find the best objective value in the first step, regardless of the consistency of the second step problem. Backtracking on the choice of the controller to find out a controller for which a corresponding prefilter exists can turn out to be difficult and time consuming. Therefore, we propose to use directly the RSM and RPT constraints in a one-step solving process. The advantage is to tackle directly the original problem, hence potentially finding optimal feasible solutions without any backtrack. The difficulty is to manage a more complex problem, where controller and prefilter design variables are involved simultaneously. Such a problem can hardly be addressed using complete global optimization methods (such as branch and bound methods), which are quite sensitive to the number of variables, but it should remain tractable for local nonlinear solvers.

The complete automation of the controller and prefilter synthesis also requires addressing two other issues of QFT problem solving: First, the robust stability of the solution is not explicitly formalized in the problem, and is typically handled manually when the problem is solved using the graphical loop-shaping approach, starting from a stable nominal controller that is manually maintained during the loop shaping. Without any stability enforcing constraints, nonlinear solvers may return unstable solutions which still satisfy stability margin and performance tracking constraints. This is tackled in Subsection 3.1. Second, QFT bounds are not well suited to nonlinear solvers, since they are given under the form of a list of points describing the boundary of the constraint. An alternative simple sampling strategy is described in Subsection 3.2.

3.1 Stability

In the usual QFT approach, the stability is usually enforced for an arbitrary nominal plant using the Nyquist stability criterion, and maintained during the loop shaping process. This raises two issues: First, although some automated procedure to compute the Nyquist stability criterion has been recently proposed in [13], it cannot be used directly within a nonlinear solver. Second, enforcing stability only for one nominal plant can turn out to be problematic in a fully automated solving process[4] (see Example 4). Therefore, we will instead use the Hurwitz stability criterion for the (sampled) uncertain plant family. It can indeed be expressed using inequality constraints, which can be handled by nonlinear solvers. This idea was already used in [19, 18] in the context of robust constraint programming, and in [8] in the context of non-robust global

[4]The stability of interval polynomials (see, e.g., [21, 39]) can also be used to prove robust stability, but considering an interval polynomial looses the dependence of its coefficients with respect to structured uncertainties, and therefore may entail some over-design to enforce stability.

optimization.

The same issue has been overcome in [25] by computing explicitly the poles and enforcing the greatest real part to be negative. Such an iterative procedure cannot be used directly in standard solvers for nonlinear programming.

3.1.1 σrelative stability through Hurwitz criterion

Hurwitz stability criterion [17] allows expressing the stability of the system in terms of (strict) inequality constraints on the coefficients of the polynomials. These constraints can be obtained formally, allowing to handle polynomials with parametric dependencies, leading to constraints on the parameters expressing the stability of the system. Here, we use the σ-relative stability, i.e., the real part of the roots of the polynomial is less that $-\sigma$ (with $\sigma > 0$), which allows enforcing a stronger stability and exploring different trad-offs. Constraints for σ-relative stability are obtained applying the Hurwitz stability criterion after having performed the change of variable $z = s + \sigma$.

Example 2 *Given the parametric polynomial $f_\gamma(s) = s^3+\gamma_2 s^2+\gamma_1 s+\gamma_0$, the Hurwitz stability constraints are:*

$$\gamma_2 > 0, \tag{20}$$
$$-\gamma_0 + \gamma_1\gamma_2 > 0, \tag{21}$$
$$\gamma_0 > 0. \tag{22}$$

The Hurwitz σ-relative stability constraints are:

$$\gamma_2 - 3\sigma > 0, \tag{23}$$
$$-\gamma_0 + \gamma_1\gamma_2 - 2\gamma_1\sigma - 2\gamma_2^2\sigma + 8\gamma_2\sigma^2 - 8\sigma^3 > 0, \tag{24}$$
$$\gamma_0 - \gamma_1\sigma + \gamma_2\sigma^2 - \sigma^3 > 0. \tag{25}$$

These latter constraints are more complex than the former, but they allow enforcing different degrees of stability. It can be proved that the three constraints (20)–(22) *entail the three constraints* (23)–(25) *for $\sigma \geq 0$, which is true in general since σ-relative stability is stronger than stability.*

Formally, given a polynomial $f_\gamma(s)$ whose coefficients depend on parameters γ, we define the Hurwitz σ-relative stability constraints $h_{\sigma,f_\gamma,i}(\gamma) > 0$, for $i \in \{0, \ldots, \deg f_\gamma\}$, as follows: First, the highest degree coefficient of f_γ has to be strictly positive. This is usually the case for the polynomials met in the context of QFT, but we nevertheless enforce it for safety by $h_{\sigma,f_\gamma,0}(\gamma) > 0$, where $h_{\sigma,f_\gamma,0}(\gamma)$ is the highest degree coefficient of f_γ (which is also the highest degree coefficient of $f_\gamma(s-\sigma)$). The degree of f_γ is therefore well defined, and the remaining constraints $h_{\sigma,f_\gamma,i}(\gamma) > 0$, for $i \in \{1, \ldots, \deg f_\gamma\}$, correspond to the constraints of the Hurwitz criterion applied to $f_\gamma(s-\sigma)$. These constraints cannot be used within local or global solvers for nonlinear programming, those latter handling only non strict inequalities. As usually done in this situation, we enforce $h_{\sigma,f_\gamma,i}(\gamma) \geq \epsilon$ instead, with $\epsilon > 0$ very small, e.g., $\epsilon = 10^{-5}$.

Assuming the prefilter is stable, we actually apply the Hurwitz σ-relative stability constraints to the denominator $D_{x,\lambda}$ of $T_{x,\lambda}$, i.e., its characteristic polynomial, which contains two kinds of parameters: The design variables x of the controller, and the uncertain plant parameters λ.

Example 3 *We consider the system of Example 1 but we limit its controller structure to one pole and one zero for simplicity. The denominator of $T_{x,\lambda}(j\omega)$ is*

$$D_{x,\lambda}(j\omega) = ak_Gk_Pp_1z_1 + (ak_Gk_Pp_1 + ap_1z_1)s + az_1s^2 + p_1z_1s^2 + z_1s^3, \tag{26}$$

where uncertain parameters are $\lambda = (a, k_P)$ and design variables are $x = (k_G, z_1, p_1)$. Then the four Hurwitz σ-relative stability constraints are:

$$h_{\sigma,f_{x,\lambda},0}(x,\lambda) = z_1 \geq \epsilon, \tag{27}$$

$$h_{\sigma,f_{x,\lambda},1}(x,\lambda) = z_1(a + p_1 - 3\sigma) \geq \epsilon, \tag{28}$$

$$\begin{aligned} h_{\sigma,f_{x,\lambda},2}(x,\lambda) = z_1(&ap_1(k_Gk_P(a+p_1) + (a - k_Gk_P + p_1)z_1)\sigma \\ &- 2(ak_Gk_Pp_1 + (a^2 + 3ap_1 + p_1^2)z_1) + 8(a+p_1)z1\sigma^2 - 8z_1\sigma^3) \geq \epsilon, \end{aligned} \tag{29}$$

$$h_{\sigma,f_{x,\lambda},3}(x,\lambda) = ak_Gk_Pp_1z_1 - ap_1(k_Gk_P + z_1)\sigma + (a+p_1)z_1\sigma^2 - z_1\sigma^3 \geq \epsilon. \tag{30}$$

3.1.2 Robust Hurwitz σrelative stability constraints

Enforcing only the stability of one nominal plant, as usually done in the QFT approach, may fail to entail the robust stability for all plant uncertainties because of potential pole/zero cancellation occurring for some uncertainties, as shown by the following example[5].

Example 4 *Consider the uncertain open loop system*

$$L_\lambda(s) = \frac{s+\lambda}{s^2 - \lambda s - 2\lambda}, \tag{31}$$

with $\lambda \in [-1, 1]$, whose closed-loop transfer function is

$$T_\lambda(s) = \frac{L_\lambda(s)}{1 + L_\lambda(s)} = \frac{s+\lambda}{(s-\lambda)(s+1)}. \tag{32}$$

This system is stable for $\lambda_0 = -1$, and satisfies robust stability margin constraints, with, e.g., $\omega_s = 1$, since $|T_\lambda(j\omega)| = \frac{1}{\sqrt{1+\omega^2}} \leq 1$ for all $\omega \in [0, +\infty]$, independently of the uncertainty. However, the system is unstable for $\lambda > 0$. This is due to a pole/zero cancellation happening for $\lambda = 0$, which prevents $|T(j\omega)|$ from becoming large enough to violate the stability margin constraint at the actual parameter value where the system goes from stable to unstable.

We adopt the following strategy to overcome this issue: We actually enforce the Hurwitz σ-relative stability constraints robustly for all parameters uncertainties:

$$\forall i \in \{0, \ldots, \deg D_{x,\lambda}\}, \forall \lambda \in \boldsymbol{\lambda},\ h_{\sigma,D_{x,\lambda},i}(x,\lambda) \leq 0, \tag{33}$$

for a given $\sigma > 0$. We call this constraint the robust Hurwitz σ-relative stability (RHS).

[5] This example seems to contradict Theorem 3.2 in [20].

3.2 Parameter sampling

The QFT bounds cannot be used efficiently in solvers dedicated to nonlinear programming, which require evaluating constraints of the form $g(x) \leq 0$ or $h(x) = 0$ and their first order derivatives. Furthermore, additional constraints like the RHS constraints cannot be translated to QFT bounds.

As a consequence, we adopt a different discretization approach: We simply select r plant samples $\widetilde{\boldsymbol{\lambda}} = \{\lambda_1, \ldots, \lambda_r\} \subseteq \boldsymbol{\lambda}$ and q frequency samples $\widetilde{\boldsymbol{\omega}} = \{\omega_1, \ldots, \omega_q\} \subseteq \boldsymbol{\omega}$, and enforce the constraints at each sample. This gives rise to qr standard differentiable nonlinear constraints for each robust constraint

$$\forall \omega \in \widetilde{\boldsymbol{\omega}}, \forall \lambda \in \widetilde{\boldsymbol{\lambda}}, \qquad |T_{x,\lambda}(j\omega)| \leq \omega_s \tag{34}$$

$$\forall \omega \in \widetilde{\boldsymbol{\omega}}, \forall \lambda \in \widetilde{\boldsymbol{\lambda}}, \quad A(\omega) \leq |F_y(j\omega)|\,|T_{x,\lambda}(j\omega)| \tag{35}$$

$$\forall \omega \in \widetilde{\boldsymbol{\omega}}, \forall \lambda \in \widetilde{\boldsymbol{\lambda}}, \quad |F_y(j\omega)|\,|T_{x,\lambda}(j\omega)| \leq B(\omega), \tag{36}$$

and $(1+\deg D_{x,\lambda})r$ standard nonlinear differentiable constraints for the RHS constraint

$$\forall i \in \{0, \ldots, \deg D_{x,\lambda}\}, \forall \lambda \in \widetilde{\boldsymbol{\lambda}}, \qquad h_{\sigma, D_{x,\lambda}, i}(x, \lambda) \geq \epsilon. \tag{37}$$

These constraints enjoy explicit expressions, which allows using them within nonlinear solvers (automatic differentiation can be used to evaluate their derivatives). The main drawback of this discretization approach is the number of generated constraints, which increases exponentially with respect to the number p of parameters λ: if each parameter is sampled s times, s^p constraints are generated. This drawback can be compensated by using more elaborate sampling strategies, e.g. in the case of linear dependences with respect to parameters, by sampling only the vertices of the parameter uncertainty box domain as proposed in [37] in the context of QFT bounds generation (the computational complexity of QFT bounds generation is also very sensitive to the number of uncertain parameters).

4 Preliminary experiments

We have proposed a novel approach to automate controller and prefilter design in single step within the QFT framework. This approach solves the optimization problem for controller and prefilter parameters that minimize a given objective function. The controller and prefilter are designed with constraints on the output (tracking specification) as well as on the robust stability margin. For these experiments, the performance objective function is chosen as a weighted sum of the gains of the controller at each design frequency:

$$c(x) = \sum_{i=1}^{q} c_i |G_x(j\omega_i)| \tag{38}$$

where ω_i are the frequency samples, the weights satisfying $c_i \geq 0$. The numerical values chosen for the experiments are 1 for the least and the greatest frequency samples, and zero for the other ones.

In this section, we take three different PID (Propotional-Integral-Derivative) controllers and prefilters synthesis problems. These examples are: (i) Well-known DC motor example [40, 30]; (ii) Lower and upper magnetic levitation system [31]; (iii) First order time delay system [32]. The nonlinear local solver FMINCON [24] has

been used. Such solvers require an initial value of variable and perform a local search. A random multi-start strategy is usually used. However, so that experiments can be reproducible, we used instead a prefilter/controller prefilter taken from the literature for each studied system as an initial starting vector. The choice of σ for the Hurwitz σ-relative stability impacts the solution: Larger σ leads to more stable solutions, but our preliminary experiments have shown stronger oscillating solution as well. This should be more investigated, but the experiments presented below use $\sigma = 0$, i.e., the simple Hurwitz stability.

The matlab code for encoding models and using FMINCON can be downloaded at `www.goldsztejn.com/src/QFT_1step.zip`.

4.1 DC motor

The uncertain plant transfer function for DC motor systems is given as

$$P_1(s) \quad = \quad \frac{ka}{s(s+a)}$$

where the uncertainties introduced in the parameters k and a are as follows:

$$\boldsymbol{\lambda}_k = [1, 10], \qquad \boldsymbol{\lambda}_a = [1, 10],$$

Samples from these plant uncertainty intervals are chosen as follows:

$$\widetilde{\boldsymbol{\lambda}}_k = \{\lambda_{k_1}, \ldots, \lambda_{k_r}\} = \{1, 3.25, 5.5, 7.75, 10\} \subseteq \boldsymbol{\lambda}_k$$

$$\widetilde{\boldsymbol{\lambda}}_a = \{\lambda_{a_1}, \ldots, \lambda_{a_r}\} = \{1, 3.25, 5.5, 7.75, 10\} \subseteq \boldsymbol{\lambda}_a$$

The design frequency set is selected as

$$\widetilde{\boldsymbol{\omega}} = \{\omega_1, \ldots, \omega_q\} = \{0.5, 1, 2, 3, 5, 10, 30, 60\} \subseteq \boldsymbol{\omega}$$

The close-loop specification of this system yields the following constraints:

- Robust stability margins(RSM):

$$\left| \frac{P_\lambda(j\omega)G_X(j\omega)}{1 + P(j\omega)G(j\omega)} \right| \leq 1.2, \qquad \forall \omega \in \widetilde{\boldsymbol{\omega}}, \lambda \in \widetilde{\boldsymbol{\lambda}}$$

- Robust tracking specification(RPT):

$$|A(j\omega)| \leq \left| F_Y(j\omega) \frac{P_\lambda(j\omega)G_X(j\omega)}{1 + P_\lambda(j\omega)G_X(j\omega)} \right| \leq |B(j\omega)|, \qquad \forall \omega \in \widetilde{\boldsymbol{\omega}}, \lambda \in \widetilde{\boldsymbol{\lambda}}$$

  where, the tracking specification uses the bounding closed-loop transfer functions:

$$A(s) = \frac{120}{s^3 + 17s^2 + 82s + 120},$$

  and

$$B(s) = \frac{0.6585(s + 30)}{s^2 + 4s + 19.752}.$$

An ideal form of PID controller and prefilter structure are selected for the design:

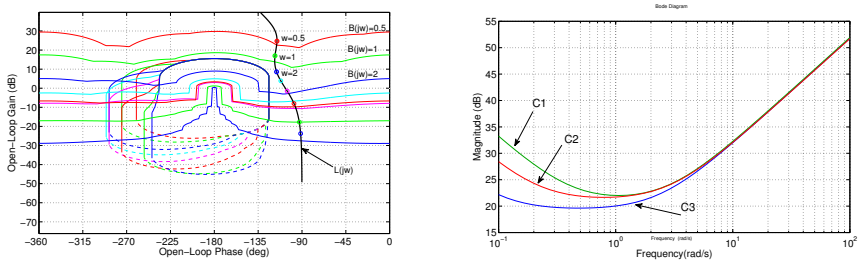
$$G_X(s) = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) \qquad F_Y(s) = \frac{1}{\left(\frac{s}{pf_1}+1\right)\left(\frac{s}{pf_2}+1\right)},$$

where $X = \{K_p, T_d, T_i\}$ is the vector of controller design parameters and $Y = \{pf_1, pf_2\}$ is the vector of prefilter design parameters. Initial domains of these parameters are considered as $K_p = [0, 100], T_d = [0, 100], T_i = [0, 100], pf_1 = [0, 100], pf_2 = [0, 100]$. In addition to this, we have chosen existing controller [40] as an initial guess of $K_{p0} = 12.6, T_{d0} = 0.31, T_{i0} = 2.83, pf_{10} = 3.7, pf_{20} = 8$.

The proposed single step approach produces the following optimum PID controller and prefilter using the local optimization solver FMINCON [24].

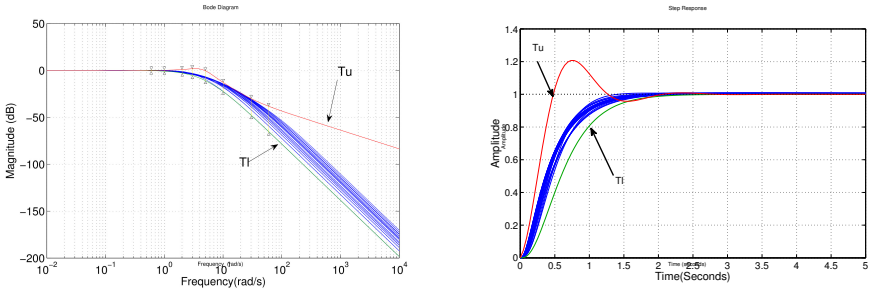
$$K_p = 9.22, \quad T_d = 0.41, \quad T_i = 12.21, \qquad pf_1 = 2.89, \quad pf_2 = 10.95$$

The Nichols chart of the nominal loop transfer function with the computed controller is shown in Figure 3 (a). It shows that the designed controller respects all the bounds at all design frequencies. Figure 3 (b) shows the comparison of the controller obtained here with the controllers proposed in the literature. It is clear from the figure that the proposed controller is less conservative in term of magnitude compared to the previously reported controllers, C1 from [40], and C2 from [30].



(a) (b)

Figure 3: (a) Nichols chart of nominal loop transfer function with designed controller for DC motor system and nominal loop transfer function (b) Comparison of bode plot of controller transfer functions between the previously proposed controllers C1 [40] and C2 [30], and the controller C3 designed using the proposed one step approach.



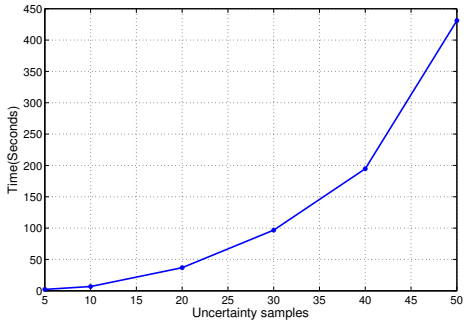
(a) (b)

Figure 4: (a) Bode plot of close loop transfer functions for different plants from plant set (b) Time domain validation with step response for different plants from plant set

Frequency domain validation and time domain validation for different uncertain plants from plant set of DC-motor are shown in Figure 4(a) and (b) respectively. One can see on Figure 4(a) that the tracking performance bounds are satisfied at all prescribed frequency samples for all considered uncertain plant samples. On the other hand, it also appears the tracking upper-bound is not respected by some uncertain plant samples in the frequency interval $[11, 29]$ which contains no frequency sample. Since the bound is not too much exceeded and most of the uncertain plant samples satisfy it, we consider that the solution is still acceptable[6].

We further analyze the limits of the uncertain parameter sampling approach we propose to handle robustness (see Section 3.2). Figure 5(a) and Table 5(b) show the graph and comparison of total number of samples versus time taken for solving the overall optimization problem, which is clearly related to the number of generated constraints since they must all be evaluated at each iteration of the local optimizer we employ. The nature of this graph is clearly quadratic in the number of samples, and exponential in the number of parameters. The solution evaluated above corresponds to the first line in Table 5(b), illustrating that the one-step approach still finds very good solutions in reasonable time.

[6]This issue is common to all techniques that use frequency samples. In case the response between samples would not be judged satisfying, it would be possible to add a few frequency samples to the specification and solve again the model. Selecting the right frequency samples to add is made easy by reading the graphic. Here, we would typically add 20.0 to the samples in $\widetilde{\boldsymbol{\omega}}$.



(a)

per param.	combinations	Time (s)
5	25	2.5120
10	100	6.7820
20	400	36.8621
30	900	96.8212
40	1600	194.6573
50	2500	431.0220

(b)

Figure 5: Number of samples vs solving time in seconds

4.2 Lower and upper magnetic levitation systems

The uncertain plant transfer function for lower and upper magnetic levitation (Maglev) systems are given as follows [31]:

Lower maglev plant:

$$P_2(s) \quad = \quad \frac{k}{s^2 + a},$$

where the uncertainties introduced in the parameters k and a are as follows:

$$\boldsymbol{\lambda}_k = [811, 944], \qquad \boldsymbol{\lambda}_a = [382, 478.5]$$

Samples from these plant uncertainty intervals are chosen as follows:

$$\widetilde{\boldsymbol{\lambda}}_k = \{\lambda_{k1}, \ldots, \lambda_{kr}\} = \{811, 877.5, 944\} \subseteq \boldsymbol{\lambda}_k$$

$$\widetilde{\boldsymbol{\lambda}}_a = \{\lambda_{a1}, \ldots, \lambda_{ar}\} = \{382., 430.25, 478.5\} \subseteq \boldsymbol{\lambda}_a$$

Upper maglev plant:

$$P_3(s) \quad = \quad \frac{k}{s^2 - a},$$

where the uncertainties introduced in the parameters k and a are as follows:

$$\boldsymbol{\lambda}_k = [1021, 1106], \qquad \boldsymbol{\lambda}_a = [382, 478.5]$$

Samples from these plant uncertainty interval are chosen as follows:

$$\widetilde{\boldsymbol{\lambda}}_k = \{\lambda_{k1}, \ldots, \lambda_{kr}\} = \{1021, 1063.5, 1106\} \subseteq \boldsymbol{\lambda}_k$$

$$\widetilde{\boldsymbol{\lambda}}_a = \{\lambda_{a1}, \ldots, \lambda_{ar}\} = \{382, 430.25, 478.5\} \subseteq \boldsymbol{\lambda}_a$$

In both cases, the design frequency set is selected as

$$\widetilde{\boldsymbol{\omega}} = \{\omega_1, \ldots, \omega_q\} = \{0.1, 1, 1.5, 2, 2.5, 3, 3.66, 5.5, 10, 20, 30\} \subseteq \boldsymbol{\omega}$$

The closed-loop specification selected for the design of controller and prefilter is as follows:

- Robust stability margins(RSM):

$$\left|\frac{P_\lambda(j\omega)G_X(j\omega)}{1+P(j\omega)G(j\omega)}\right| \leq 1.2, \qquad \forall\omega \in \widetilde{\boldsymbol{\omega}}, \lambda \in \widetilde{\boldsymbol{\lambda}}$$

- Robust tracking specification(RPT):

$$|A(j\omega)| \leq \left|F_Y(j\omega)\frac{P_\lambda(j\omega)G_X(j\omega)}{1+P_\lambda(j\omega)G_X(j\omega)}\right| \leq |B(j\omega)|, \qquad \forall\omega \in \widetilde{\boldsymbol{\omega}}, \lambda \in \widetilde{\boldsymbol{\lambda}}$$

where, the tracking specification uses the bounding closed-loop transfer functions:

$$A(s) = \frac{916.3}{s^3+39.76s^2+354.9s+916.3},$$

and

$$B(s) = \frac{1.722s+68.89}{s^2+16.6s+68.89}.$$

An ideal form of PID controller and prefilter structure are selected for the design.

$$G_X(s) = K_p\left(1+\frac{1}{T_i s}+T_d s\right) \qquad F_Y(s) = \frac{1}{\left(\frac{s^2}{\omega_n^2}+\frac{2\zeta s}{\omega_n}+1\right)}.$$

where $\{K_p, T_d, T_i, \omega_n, \zeta\}$ is the vector of controller and prefilter design parameters. Initial domains of these parameters are considered as $K_p = [0,100], T_d = [0,100], T_i = [0,100], \omega_n = [0,100], \zeta = [0,1]$. In addition to this, we have chosen existing controller [31] as an initial guess of $K_{p0} = 1.16, T_{d0} = 0.05, T_{i0} = 0.25, \omega_{n0} = 7.6, \zeta_0 = 0.81$ for lower Maglev system and $K_{p0} = 4.98, T_{d0} = 0.04, T_{i0} = 0.57, \omega_{n0} = 4.85, \zeta_0 = 0.84$ for upper Maglev system.

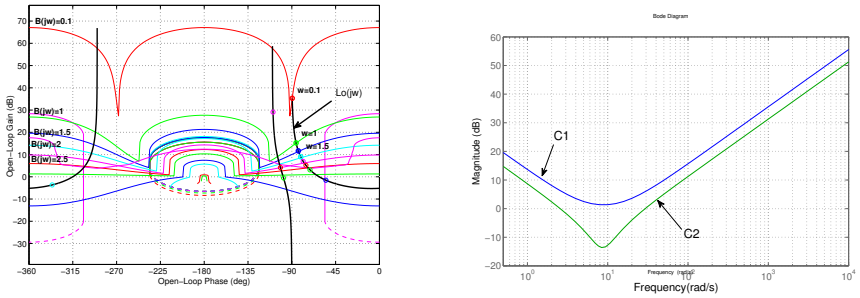
The proposed single step approach produces the optimum PID controller and prefilter using local optimization solver FMINCON [24].

The parameters of PID controller and prefilter for the lower Maglev system are

$$K_p = 0.2085, \quad T_d = 0.1757, \quad T_i = 0.0760, \qquad w_n = 7.7990, \quad \zeta = 0.5087$$

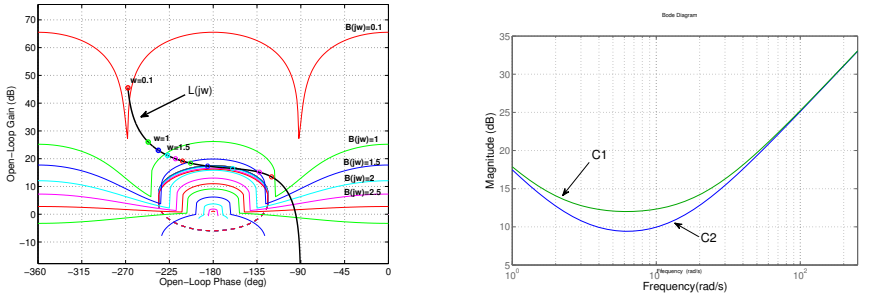
The parameters of PID controller and prefilter for the upper Maglev system are

$$K_p = 2.956, \quad T_d = 0.0604, \quad T_i = 0.4195, \qquad w_n = 5.0075, \quad \zeta = 0.9950$$



(a) (b)

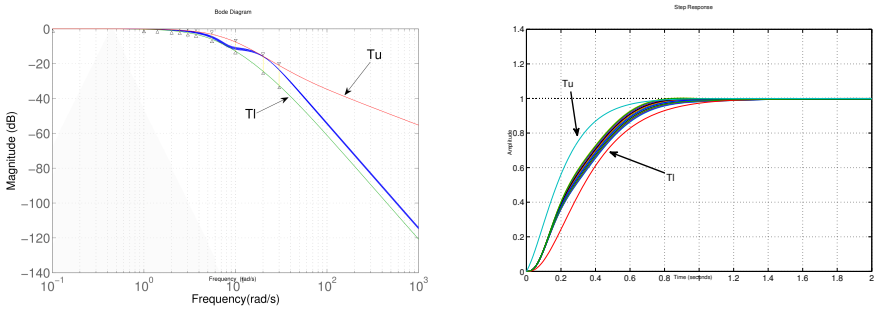
Figure 6: Lower Maglev Plant: (a) Nichols chart of nominal loop transfer function with designed controller for lower magnetic system and nominal loop transfer function (b) Comparison of bode plot of controller transfer function with existing controller C1 [31] and controller C2 designed by proposed one step approach



(a) (b)

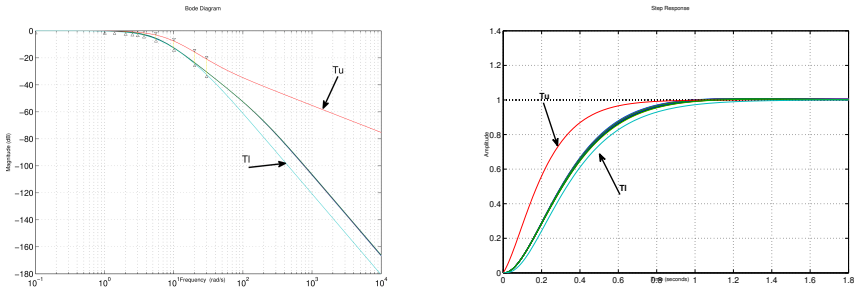
Figure 7: Upper Maglev plant: (a)Nichols chart of nominal loop transfer function with designed controller (b) Comparison of bode plot of nominal loop transfer function with existing controller and controller designed by proposed one step approach

The Nichols chart of the nominal loop transmission function with the computed controller is shown in Figures 6(a) and 7(a). It shows that the designed controller respects all the bounds at all design frequencies. Figures 6(b) and 7(b) show the comparison with existing controllers from [31] and the controller obtained using the single step approach. It is clear from the figure that the gains of the obtained controllers are less than that of the existing controllers at different design frequencies.



(a) (b)

Figure 8: Lower Maglev Plant: (a) Bode plot of close loop transfer functions for different plants from plant set (b) Time domain validation for different plants from plant set



(a) (b)

Figure 9: Upper Maglev plant: (a) Bode plot of close loop transfer functions for different plants from plant set (b) Time domain validation for different plants from plant set

Frequency domain validations and time domain validations for different uncertain plants from plant sets of lower and upper magnetic levitation plants are shown in Figure 8 and in Figure 9 respectively.

4.3 Time Delay System

In real life, many processes are modeled as time delay systems. The synthesis of controllers for such systems is really a difficult task. An automated synthesis for PID controllers using proposed method is presented in this section.

First order plant with time delay model is represented as follows [32]:

$$P_4(s) = \frac{k}{s+a}e^{-t_d s},$$

By taking the Pade approximation for the delay term, this system can also be represented as:

$$P_4(s) = \frac{k(1 - \frac{t_d s}{2})}{(s+a)(1 + \frac{t_d s}{2})}, \tag{39}$$

where the uncertainties introduced in the parameters k , a and t_d are as follows:

$$\boldsymbol{\lambda}_k = [1, 3], \quad \boldsymbol{\lambda}_a = [1, 2], \quad \boldsymbol{\lambda}_{t_d} = [0.08, 0.12]$$

Here, nominal value for time delay is 0.1 and we consider $\pm 20\%$ uncertainties in the nominal time delay value.

Samples from these plant uncertainty intervals are chosen as follows:

$$\widetilde{\boldsymbol{\lambda}}_k = \{\lambda_{k1}, \ldots, \lambda_{kr}\} = \{1 \;\; 2 \;\; 3\} \subseteq \boldsymbol{\lambda}_k$$

$$\widetilde{\boldsymbol{\lambda}}_a = \{\lambda_{a1}, \ldots, \lambda_{ar}\} = \{1 \;\; 1.5 \;\; 2\} \subseteq \boldsymbol{\lambda}_a$$

$$\widetilde{\boldsymbol{\lambda}}_{t_d} = \{\lambda_{t_{d1}}, \ldots, \lambda_{t_{dr}}\} = \{0.08 \;\; 0.1 \;\; 0.12\} \subseteq \boldsymbol{\lambda}_{t_d}$$

The design frequency set is selected as

$$\widetilde{\boldsymbol{\omega}} = \{\omega_1, \ldots, \omega_q\} = \{0.1, 0.2, 0.5, 1, 2, 5, 8, 10, 50\} \subseteq \boldsymbol{\omega}$$

The closed-loop specification selected for the design of controller and prefilter is as follows:

- Robust stability margins:

$$\left| \frac{P_\lambda(j\omega) G_X(j\omega)}{1 + P_\lambda(j\omega) G_X(j\omega)} \right| \leq 1.2, \qquad \forall \omega \in \widetilde{\boldsymbol{\omega}}, \lambda \in \widetilde{\boldsymbol{\lambda}}$$

- Robust tracking specification:

$$|A(j\omega)| \leq \left| F_Y(j\omega) \frac{P_\lambda(j\omega) G_X(j\omega)}{1 + P_\lambda(j\omega) G_X(j\omega)} \right| \leq |B(j\omega)|, \qquad \forall \omega \in \widetilde{\boldsymbol{\omega}}, \lambda \in \widetilde{\boldsymbol{\lambda}}$$

  where, the tracking specification uses the bounding closed-loop transfer functions:

$$A(s) = \frac{9}{s^3 + 7s^2 + 15s + 9},$$

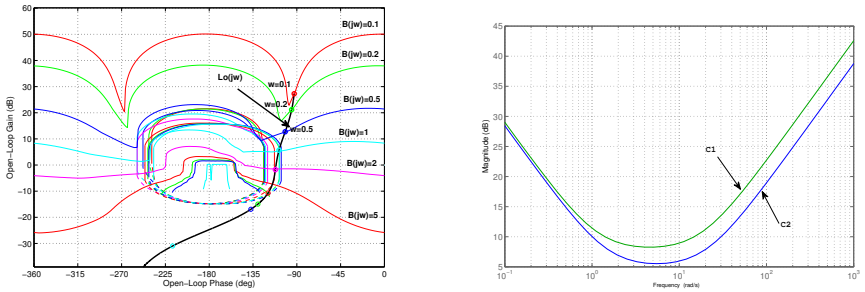
  and

$$B(s) = \frac{4}{s^2 + 3.3s + 4}.$$

An ideal form of PID controller and prefilter structure and initial domains are considered as in section 4.1. We have chosen existing controller [32] as an initial guess of $K_{p0} = 2.58, T_{d0} = 0.05, T_{i0} = 0.92, pf_1 = 1.40, pf_2 = 2.90$.

The proposed single step approach produces the optimum PID controller and prefilter using local optimization solver FMINCON [24]:

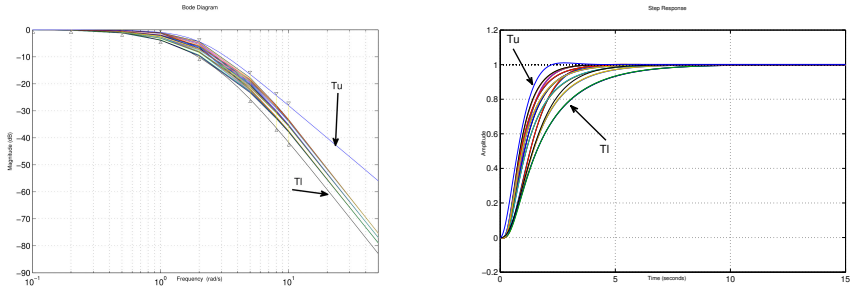
$$K_p = 1.88, \quad T_d = 0.05, \quad T_i = 0.72, \qquad pf1 = 2.13, \quad pf2 = 2.13$$



(a) (b)

Figure 10: (a) Nichols chart of nominal loop transfer function with designed controller for first order time delay system (b) Comparison of bode plot of controller transfer functions with PID controller from the literature (C1) and PI controller designed by proposed one step approach (C2).

The nominal loop transmission function with the obtained controller is shown in Figure 10. It shows that the designed controller respects all the bounds at all design frequencies. It is interesting to notice that the proposed approach automatically removes one of the redundant design parameter and produces an optimum PI controller instead of a PID. Figure 10 (b) shows the comparison with existing controller [32], illustrating the much better gain of the obtained controller at all frequencies.



(a) (b)

Figure 11: (a) Bode plot of close loop transfer functions for different plants from plant set (b) Time domain validation for different plants from plant set

The performance of the controller and prefilter obtained are verified in frequency domain as well as in time domain. Frequency domain verification is shown in Figure 11(a) Time domain simulation result of the different plants from plant set are shown in Figure 11(b).

4.4 Flight control system

Here, we have considered another QFT benchmark example to test the quality of the new methodology. The uncertain transfer function for flight control problem is given as follows [25]

$$P_5(s) \quad = \quad \frac{k(1+\frac{s}{z})}{s(1+\frac{s}{p})(1+\frac{2\zeta s}{\omega_n}+\frac{s^2}{\omega_n^2})}$$

where the uncertainties introduced in the parameters k , a and t_d are as follows:

$$\boldsymbol{\lambda}_k = [0.2, 2], \quad \boldsymbol{\lambda}_z = [0.5, 0.75], \quad \boldsymbol{\lambda}_p = [1, 10], \quad \boldsymbol{\lambda}_{\omega_n} = [5, 6], \boldsymbol{\lambda}_\zeta = [0.8, 0.9]$$

Samples from these plant uncertainty intervals are chosen as follows:

$$\widetilde{\boldsymbol{\lambda}}_k = \{\lambda_{k_1}, \ldots, \lambda_{k_r}\} = \{0.2,\ 1,\ 2\} \subseteq \boldsymbol{\lambda}_k,\ \widetilde{\boldsymbol{\lambda}}_z = \{\lambda_{z_1}, \ldots, \lambda_{z_r}\} = \{0.5,\ 0.6,\ 0.75\} \subseteq \boldsymbol{\lambda}_z$$

$$\widetilde{\boldsymbol{\lambda}}_p = \{\lambda_{p_1}, \ldots, \lambda_{p_r}\} = \{1,\ 5,\ 10\} \subseteq \boldsymbol{\lambda}_p,\ \widetilde{\boldsymbol{\lambda}}_{\omega_n} = \{\lambda_{\omega_{n_1}}, \ldots, \lambda_{\omega_{n_r}}\} = \{5,\ 5.5,\ 6\} \subseteq \boldsymbol{\lambda}_a$$

$$\widetilde{\boldsymbol{\lambda}}_\zeta = \{\lambda_{\zeta_1}, \ldots, \lambda_{\zeta_r}\} = \{0.8,\ \ 0.85,\ \ 0.9\} \subseteq \boldsymbol{\lambda}_\zeta$$

The design frequency vector is selected as

$$\begin{aligned}\widetilde{\boldsymbol{\omega}} = \{\omega_1, \ldots, \omega_q\} = &\{0.01,\ 0.02,\ 0.05,\ 0.1,\ 0.2,\ 0.5,\ 1\ 2,\ 3,\ 4,\\ &5,\ 7,\ 8,\ 9,\ 10,\ 20,\ 30,\ 40,\ 50,\ 80,\ 100,\ 300\} \subseteq \boldsymbol{\omega}\end{aligned}$$

The closed-loop specification selected for the design of controller and prefilter is as follows:

- Robust stability margins:

$$\left|\frac{P_\lambda(j\omega)G_X(j\omega)}{1+P_\lambda(j\omega)G_X(j\omega)}\right| \leq 6\ dB, \qquad \forall \omega \in \widetilde{\boldsymbol{\omega}}, \lambda \in \widetilde{\boldsymbol{\lambda}}$$

- Robust tracking specification:

$$|A(j\omega)| \leq \left|F_Y(j\omega)\frac{P_\lambda(j\omega)G_X(j\omega)}{1+P_\lambda(j\omega)G_X(j\omega)}\right| \leq |B(j\omega)|, \qquad \forall \omega \in \widetilde{\boldsymbol{\omega}}, \lambda \in \widetilde{\boldsymbol{\lambda}}$$

  where, the tracking specification uses the bounding closed-loop transfer functions:

$$A(j\omega) = \frac{(1+\frac{j\omega}{0.35})}{(1+\frac{j\omega}{3})(1+\frac{j\omega}{0.5})},$$

  and

$$B(j\omega) = \frac{1}{(1+j\omega)(1+j\omega)(1+\frac{j\omega}{2})}.$$

In this example, we have considered general pole-zero structure for controller and prefilter as

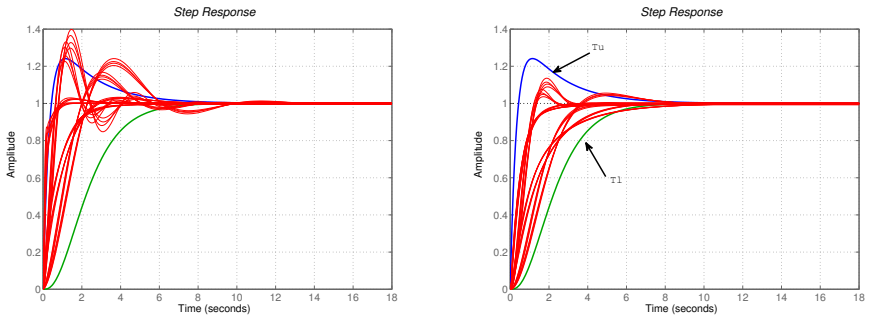
$$G_X(s) = \frac{b_4 s^4 + b_3 s^3 + b_2 s^2 + b_1 s + b_0}{s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \qquad F_Y(s) = \frac{1}{(1+pf_1)}$$

where $\{b_4, b_3, b_2, b_1, b_0, a_3, a_2, a_1, a_0, pf_1\}$ is the vector of controller and prefilter design parameters. Initial domains of all these parameters are considered as $[0, 1e8]$. We

have chosen existing controller [25] as an initial guess of $b_4 = 62.77, b_3 = 1099.79, b_2 = 6911.4970, b_1 = 18772.59, b_0 = 18354.41, a_3 = 117.07, a_2 = 3378.96, a_1 = 7009.13, a_0 = 2861.76, pf_1 = 3.5$ for flight control system.

The proposed single step approach produces the controller and prefilter using local optimization solver FMINCON [24]:

$$b_4 = 28.93,\ b_3 = 1129.58,\ b_2 = 5134.47,\ b_1 = 13824.02,\ b_0 = 3691.50,$$
$$a_3 = 54.31,\ a_2 = 3016.39,\ a_1 = 4414.70,\ a_0 = 804.14 \quad pf_1 = 3.04$$



(a) (b)

Figure 12: (a) Time domain validation for different plants from plant set with existing controller [25] (b) Time domain validation for different plants from plant set with proposed one step approach

The performance of obtained controller in time domain is presented in Figure 12 The comparison with existing controller clearly shows that proposed controller and prefilter pair provides better robustness to parameter uncertainties.

5 Conclusion

A new simple and automated method is presented for designing robust optimal controller and prefilter for uncertain systems. It satisfies all robust design specification for stability as well as tracking at all design frequencies. Unlike all QFT approaches until now, the proposed method combines the controller synthesis and the prefilter synthesis into a single step. This eliminates the need to backtrack to the controller synthesis step when there is no prefilter for the controller chosen in the first step. This removes the trial-and-error component of the classical loop-shaping approaches, which is usually involved in controller and prefilter synthesis steps. We claim our approach opens the way towards less conservative design then previous approaches.

Stability criteria which can be encoded as numerical constraints are also proposed in order to allow using standard nonlinear solvers for a fully automated synthesis. A new objective function helps to find better controllers with lower gain at each design frequencies. The proposed method has been successfully applied to solve a variety of

test cases, including, the benchmark DC-motor plant, a resonant plant, an open loop unstable plant and a plant with variation in the time delay.

This research was supported by the Indo-French CEFIPRA project under DST (number 4502-1).

References

[1] M. Araki and H. Taguchi. Two-degree-of-freedom PID controllers. *International Journal of Control, Automation, and Systems*, 1(4):401–411, 2003.

[2] I. Araya, G. Trombettoni, B. Neveu, and G. Chabert. Upper bounding in inner regions for global optimization under inequality constraints. *Journal of Global Optimization*, 60(2):145–164, 2014.

[3] C. Borghesani, Y. Chait, and O. Yaniv. *The Quantitative Feedback Theory toolbox for MATLAB*. The MathWorks, Inc., MA, USA, 1995.

[4] Jose Carlos Moreno, Alfonso Banos, and Manuel Berenguel. Improvements on the computation of boundaries in QFT. *International Journal of Robust and Nonlinear Control*, 16(12):575–597, 2006.

[5] J. Cervera and A. Banos. Automatic loop shaping in QFT using CRONE structures. *JVC/Journal of Vibration and Control*, 14(9-10):1513–1529, 2008.

[6] G. Chabert and L. Jaulin. Contractor Programming. *Artificial Intelligence*, 173(11):1079–1100, 2009.

[7] Y. Chait and O. Yaniv. Multi-input/single-output computer-aided control design using the quantitative feedback theory. *International Journal of Robust and Nonlinear Control*, 3(1):47–54, 1993.

[8] YoungJung Chang and NikolaosV. Sahinidis. Global optimization in stabilizing controller design. *Journal of Global Optimization*, 38(4):509–526, 2007.

[9] W. H. Chen, D. J. Ballance, W. Feng, and Y. Li. Genetic algorithm enabled computer-automated design of qft control systems. In *Computer Aided Control System Design, 1999. Proceedings of the 1999 IEEE International Symposium on*, pages 492–497, 1999.

[10] W. H. Chen, D. J. Ballance, and Y. Li. Automatic loop-shaping in QFT using genetic algorithms. In *3rd Asia-Pacific Conference on Control and Measurement*, pages 63–67, 1998.

[11] P. D. Dorato. A historical review of robust control. *IEEE Control Systems Magazine*, 7(2):44–47, 1987.

[12] Jorge Elso, Montserrat Gil-Martinez, and Mario Garcia-Sanz. Nonconservative QFT bounds for tracking error specifications. *International Journal of Robust and Nonlinear Control*, 22(18):2014–2025, 2012.

[13] Mario Garcia-Sanz. The nyquist stability criterion in the nichols chart. *International Journal of Robust and Nonlinear Control*, pages n/a–n/a, 2015.

[14] Mario Garcia-Sanz and Constantine H. Houpis. *Wind Energy Systems: Control Engineering Design*. CRC Press, Taylor & Francis, 2012.

[15] A. Gera and Horowitz I. M. Optimization of the loop transfer function. *International Journal of Control*, 31(2):389–398, 1980.

[16] L. Granvilliers and F. Benhamou. Algorithm 852: Realpaver: an Interval Solver using Constraint Satisfaction Techniques. *ACM TOMS*, 32(1):138–156, 2006.

[17] A. Hurwitz. Ueber die bedingungen, unter welchen eine gleichung nur wurzeln mit negativen reellen theilen besitzt. *Mathematische Annalen*, 46(2):273–284, 1895.

[18] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. *Applied Interval Analysis with Examples in Parameter and State Estimation, Robust Control and Robotics*. Springer-Verlag, 2001.

[19] L. Jaulin and E. Walter. Guaranteed tuning, with application to robust control and motion planning. *Automomatica*, 32(8):1217–1221, 1996.

[20] Suhada Jayasuriya and Yongdong Zhao. Robust stability of plants with mixed uncertainties and quantitative feedback theory. *Journal of Dynamic Systems, Measurement, and Control*, 116(1):10–16, 1994.

[21] V.L. Kharitonov. Asymptotic stability of an equilibrium position of a family of systems of linear differential equations. *Differ. Uravn.*, pages 1483–1485, 1978.

[22] Horowitz I. M. *Quantitative Feedback Design Theory (QFT)*. QFT Publication, 1993.

[23] Juan Jose Martin-Romero, Montserrat Gil-Martinez, and Mario Garcia-Sanz. Analytical formulation to compute QFT bounds: The envelope method. *International Journal of Robust and Nonlinear Control*, 19(17):1959–1971, 2009.

[24] MATLAB. *version 8.5.0(R2015a)*. Natick, Massachusetts, 2015.

[25] C. Molins and M. Garcia-Sanz. Automatic loop-shaping of qft robust controllers. In *Proceedings of the IEEE 2009 National Aerospace Electronics Conference (NAECON)*, pages 103–110, July 2009.

[26] P.S.V. Nataraj. Computation of QFT bounds for robust tracking specifications. *Automatica*, 38(2):327 – 334, 2002.

[27] P.S.V. Nataraj and M.M. Deshpande. Automated synthesis of fixed structure QFT controller using interval constraint satisfaction techniques. *In Proceedings of the* 17^{th} *IFAC World Congress*, pages 4976–4981, July 2008.

[28] P.S.V. Nataraj and M.M. Deshpande. Implementation of fixed structure QFT prefilter synthesized using interval constraint satisfaction technique. *International Journal of Reliability and Safety*, 6:255–281, 2012.

[29] P.S.V. Nataraj and S.S. Tharewal. Automatic design of QFT prefilter using interval analysis. *IEEE International Symposium on Computer Aided Control System Design*, 14:156–160, 2004.

[30] P.S.V. Nataraj and S.S. Tharewal. An interval analysis algorithm for automated controller synthesis in QFT design. *Trans. of the ASME Journal of Dynamic System, Measurement and Control*, 129:311–321, 2007.

[31] M. D. Patil, P. S. V. Nataraj, and V. A. Vyawahare. Real time implementation of robust qft controller and prefilter for magnetic levitation system. In *Industrial Instrumentation and Control (ICIC), 2015 International Conference on*, pages 703–708, May 2015.

[32] H. Purohit and P. S. V. Nataraj. Optimized and automated synthesis of robust pid controller with quantitative feedback theory. In *Industrial Instrumentation and Control (ICIC), 2015 International Conference on*, pages 105–110, May 2015.

[33] Nataraj S. V. Paluri and Nandkishor Kubal. Automatic loop shaping in QFT using hybrid optimization and constraint propagation techniques. *International Journal of Robust and Nonlinear Control*, 17(2-3):251–264, 2007.

[34] D. F. Thompson and O. D. I. Nwokah. Analytical loop shaping methods in Quantitative Feedback Theory. *Trans. of the ASME Journal of Dynamic Sytems, Measurement and Control*, 116:169–177, 1994.

[35] Gilles Trombettoni, Ignacio Araya, Bertrand Neveu, and Gilles Chabert. Inner Regions and Interval Linearizations for Global Optimization. In Wolfram Burgard and Dan Roth, editors, *AAAI*. AAAI Press, 2011.

[36] Q. Chen Y. Chait and C. V. Hollot. Automatic loop-shaping of QFT controllers via linear programming. *Trans. of the ASME Journal of Dynamic Systems, Measurement and Control*, 121:351–357, 1999.

[37] Shih-Feng Yang. Generation of QFT bounds for robust tracking specifications for plants with affinely dependent uncertainties. *International Journal of Robust and Nonlinear Control*, 21(3):237–247, 2011.

[38] O. Yaniv. Automatic loop shaping of MIMO controllers satisfying sensitivity specifications. *Journal of Dynamic Systems, Measurement and Control*, 128:463–471, 2006.

[39] Y. Zhao and S. Jayasuriya. A frequency domain criterion for robustly stabilizing a family of interval plants. In *American Control Conference, 1992*, pages 1403–1407, 1992.

[40] A. C. Zolotas and G. D. Halikias. Optimal design of PID controllers using the QFT method. *IEEE Proceedings, Control Theory and Application*, 146(6):585–589, 1999.