

A NEW EDUCATIONAL SOFTWARE TOOL FOR ROBUST CONTROL DESIGN USING THE QFT METHOD

R. Nandakumar and G.D. Halikias

School of Engineering and Mathematical Sciences, City University, London EC1V, UK, {et708@city.ac.uk, g.halikias@city.ac.uk}

A.C. Zolotas

Electronic and Electrical Engineering, Imperial College London, London, SW7, UK, {a.zolotas@ieee.org}

Keywords: Robust Control, Quantitative Feedback Theory (QFT), Loop-shaping, Control Systems Computer-Aided Design (CSCAD), Educational software

ABSTRACT

We present a new educational software tool for robust control design based on the Quantitative Feedback Design (QFT) method. This is a graphical design methodology for systems with large parametric uncertainty, which has been successfully applied to many complex practical problems. The software tool is implemented in Matlab and may be used to introduce students to robust control methods via small and medium-size design applications. The software is a library of programmable M-files with open access to users and is intended as a test-bed for developing new techniques in this area and for automating parts of the design procedure, such as loop-shaping. A simple design problem is used to illustrate the main features of the software.

1. INTRODUCTION

Quantitative Feedback Theory (QFT) is a systematic robust control design methodology for systems subject to large parametric or unstructured uncertainty. QFT is a graphical loop-shaping procedure, traditionally carried out on the Nichol’s chart, which can be used for the control design of either SISO or MIMO uncertain systems, including the non-linear and time-varying cases [5, 8, 11]. The main advantage of QFT relative to alternative design methodologies is that it relies on classical-control concepts and employs simple loop-shaping techniques. Additional advantages include: (i) The ability to assess quantitatively the “cost of feedback” [6], (ii) the ability to use phase-uncertainty information, and (iii) the ability to provide “transparency” in the design, i.e. clear trade-off criteria between controller complexity and the feasibility of the design objectives. Note that (iii) implies in practice that QFT often results in simple controllers which are easy to implement.

The QFT design procedure is based on the two-degree of freedom feedback configuration shown in Figure 1. In this diagram $G(p, s)$ denotes the uncertain plant, while $K(s)$ and $F(s)$ denote the feedback compensator and pre-filter, respectively, which are to be designed. Note that model uncertainty is described by the r -parameter vector $p \in \mathcal{P} \subseteq \mathbb{R}^r$ taking values in the set \mathcal{P} ; it is further assumed that $G(p, s)$ has the same number of RHP poles for all

$p \in \mathcal{P}$. Translating the uncertainty into the frequency domain, gives rise to the plant’s “uncertainty templates” which are the sets:

$$\mathcal{G}_\omega = \{G(p, j\omega) : p \in \mathcal{P}\}$$

For each fixed frequency ω , \mathcal{G}_ω defines a “fuzzy region” on the Nichol’s chart which describes the uncertainty of the plant at frequency ω in terms of magnitude (in dB’s) and phase (in degrees). For design purposes, we construct N uncertainty templates corresponding to a discrete set of frequencies $\{\omega_1, \omega_2, \dots, \omega_N\}$ chosen to cover adequately the system’s bandwidth.

The robust performance objectives of the design include good tracking of reference input $r(s)$ and good attenuation of the disturbance signal $d(s)$ entering at the system’s output, despite the presence of uncertainty. The robust tracking objectives are captured by the set of inequalities:

$$\max_{p \in \mathcal{P}} \Delta \left| \frac{G(p, j\omega_i)K(j\omega_i)}{1 + G(p, j\omega_i)K(j\omega_i)} \right|_{\text{dB}} \leq \delta(\omega_i)$$

$$:= B_u(\omega_i)|_{\text{dB}} - B_l(\omega_i)|_{\text{dB}}$$

for each $i = 1, 2, \dots, N$, i.e. if, for each frequency ω_i , the maximum variation in closed loop gain as $p \in \mathcal{P}$ does not exceed the maximum allowable spread in specifications $\delta(\omega_i)$, typically specified via two appropriate magnitude frequency responses $B_u(\omega) = |B_u(j\omega)|$ and $B_l(\omega) = |B_l(j\omega)|$. Note that it is not necessary to bound the actual gain (but only the gain spread) since we assume that, (i) no uncertainty is associated with the feedback controller $K(s)$, and (ii) the pre-filter $F(s)$ can provide arbitrary scaling to the closed-loop gain. The robust disturbance-rejection

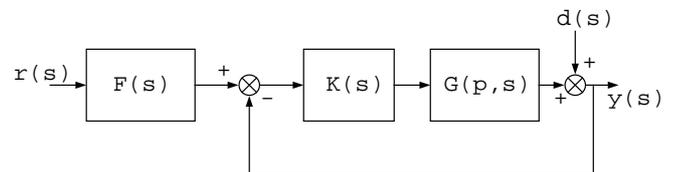


Figure 1: Feedback Configuration

objective can be satisfied by bounding the sensitivity function, i.e. by imposing constraints of the form

$$\max_{p \in \mathcal{P}} \left| \frac{1}{1 + G(p, j\omega_i)K(j\omega_i)} \right| \leq D(\omega_i)$$

for a (subset) of the design frequencies $\{\omega_1, \omega_2, \dots, \omega_N\}$. Again these are typically specified via an appropriate magnitude frequency-response $D(\omega) = |D(j\omega)|$.

Robust stability is enforced by ensuring that: (i) no unstable pole-zero cancellations occur between the plant and the controller (for every $p \in \mathcal{P}$), (ii) the nominal open-loop frequency response $L_o(j\omega) = G(p_o, j\omega)K(j\omega)$ (defined for any $p_o \in \mathcal{P}$) does not cross the -1 point (i.e. the $(-180^\circ, 0)$ point on Nichol's chart) and makes a total number of (anti-clockwise) encirclements around it equal to the number of unstable poles of $L_o(s) = G(p_o, s)K(s)$, and (iii) That no (perturbed) open-loop response crosses the -1 point, i.e.

$$-1 \notin \bigcup_{\omega \in \mathbb{R}} K(j\omega)\mathcal{G}_\omega$$

Note that condition (i) is automatically satisfied if $K(s)$ is restricted to be stable and minimum-phase, while conditions (ii) and (iii) can be easily tested graphically [4, 3]. In practice, a more severe condition than (iii) is imposed: To establish a minimum amount of damping, it is required that the nominal open-loop frequency response does not penetrate a closed contour in the Nichol's chart (\mathcal{U} -contour); this is constructed from an appropriate M -circle and information about high-frequency uncertainty of the plant [5, 8].

2. QFT SOFTWARE DESIGN TOOL

In this section we outline the main features of the developed software tool. We illustrate some of these features with a simple “benchmark” problem taken from [9]. The uncertain plant $G(p, s)$ is defined as $G(p, s) = \frac{ka}{s(s+a)}$ with $\mathcal{P} = \{(a, k) : 1 \leq a \leq 10, 1 \leq k \leq 10\}$. For simplicity, only robust tracking bounds are considered, with

$$B_u(s) = \frac{0.6585(s+30)}{s+(2 \pm j3.969)}, \text{ and } B_l(s) = \frac{3520}{(s+4)^2(s+4.4)(s+50)}$$

Seven design frequencies have been selected, i.e. $\{\omega_i\} = \{0.5, 1, 2, 3, 5, 15, 100\}$ rads/s. The M -circle defining the \mathcal{U} -contour was taken as $M = 1.2$. The design proceeds via the following steps:

2.1 Uncertainty templates

For the purposes of the design, system uncertainty needs to be translated from parameter-space to the frequency-domain (Nichol's chart). This is normally performed by discretising the uncertainty intervals for each unknown parameter and calculating the frequency response for all possible combinations at each design frequency, giving rise to a number of “fuzzy” regions in Nichol's chart (“uncertainty templates”). Although this method is clearly impractical for highly complex systems it works reasonably well in practice for small/medium-complexity problems, i.e. when the number of independently varying parameters is small (no more than 5 say).

To reduce the complexity of the QFT algorithm, a computational routine may be used at this stage to substitute the templates by their convex hull. In many cases this can be performed without introducing significant conservativeness in the design.

The uncertainty templates of the system given in the above example at the seven design frequencies are plotted in Figure 2, along with their computed convex hull. Additional graphical tools have also implemented

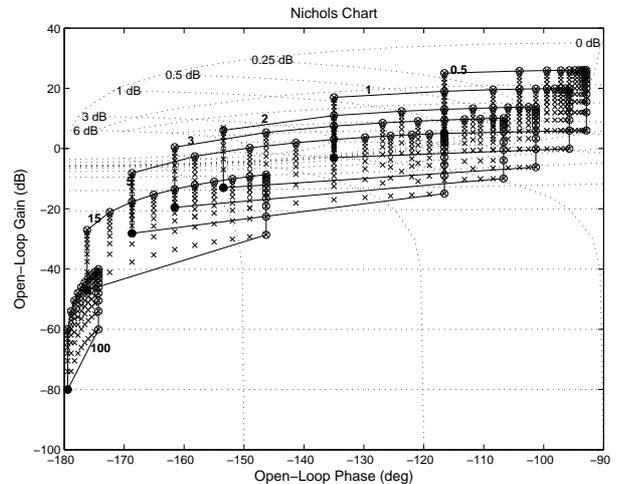


Figure 2: Uncertainty Templates with Convex Hull

for displaying multiple frequency responses on the Nichol's chart, defining a grid of M and N circles on the chart with default or user-specified values, etc.

2.2 Generation of Horowitz bounds and \mathcal{U} -contour

The robust tracking and disturbance rejection objectives have been formulated as gain inequalities of the closed-loop transfer functions (sensitivity and complementary sensitivity) at the design frequencies. For the purposes of QFT design, these inequalities must be translated into constraints on the nominal open-loop response $L_o(j\omega)$. This procedure results into a number of contours (“Horowitz templates” $f_i^t(\phi)$ and “disturbance-rejection templates” $f_i^d(\phi)$) for each frequency ω_i , $i = 1, 2, \dots, N$; these are functions of the phase variable $\phi \in (-360^\circ, 0^\circ]$. Thus, robust tracking is satisfied at frequency ω_i if $|L_o(j\omega_i)|_{\text{dB}} \geq f_i^t(\phi_i)$ where $\arg L_o(j\omega_i) = \phi_i$; similarly, robust disturbance-rejection is attained at frequency ω_i if $|L_o(j\omega_i)|_{\text{dB}} \geq f_i^d(\phi)$.

The routines for generating the robust tracking and disturbance-rejection templates are based on a simple bisection algorithm over a suitably discretised phase grid. In the case of tracking bounds, the phase of the nominal open-loop system is first fixed, and the open-loop gain is varied until the maximum variation in closed-loop gain over the corresponding uncertainty template is equal (within a small specified gain tolerance) to the “spread” in closed-loop specifications $\delta(\omega_i)$. Particular care must be taken when the template encloses the critical (-1) point, since uncertainty templates are normally specified by points at their boundary. (This condition may be easily checked by a simple routine that estimates the angle variation of the vector connecting the critical point to a point on the boundary of the template). Repeating the procedure over the discretised phase grid at each frequency

of interest results in N Horowitz templates (one for each design frequency), specifying the minimum open-loop gain required to meet the robust tracking specifications at each frequency. A similar approach is used to construct the robust disturbance-rejection bounds at the N frequencies of interest. Note that the complete procedure for generating the robust-performance templates is completely automated and need not be visible to the user. Since their generation is fairly complex computationally, these are calculated only once and saved in the workspace (or the disk) for the remaining parts of the design, unless the performance specifications need to be altered.

The \mathcal{U} -contour is obtained automatically by extending vertically the lower part of the M -circle specified by the designer by an amount equal to the gain-spread between the maximum-gain point and the nominal point in the uncertainty template at a very high (theoretically infinite) frequency. (Note that under the assumption that the plant is only subject to parametric uncertainty, the uncertainty templates tend to vertical lines at high frequencies). The \mathcal{U} -contour approach, originally proposed in [7], is a simple way of shaping the high-frequency characteristics of the system, although clearly restrictive since it does not take into account the typical loss of phase information at high frequencies (e.g. due to unmodelled high-frequency dynamics). Future versions of the software will address the problem of mixed types of uncertainty (both structured and unstructured).

Having obtained the robust-performance (tracking, disturbance-rejection) and robust stability bounds (\mathcal{U} -contour) the overall bounds of the design can be calculated, by combining appropriately the individual bounds for each point of the phase-grid. Again, this is a purely automatic procedure which need not be visible to the viewer. If the design of the feedback controller is performed manually, the overall bounds need to be clearly displayed on the Nichol's chart. Alternatively, if an optimisation technique is used to design the feedback controller, the overall bounds may need to be translated to appropriate mathematical relations (e.g. linear or nonlinear inequalities) which are then automatically supplied to the optimisation algorithm at the next phase of the design.

2.3 Design of feedback controller (loop-shaping)

The design of the feedback controller is the most important step of the QFT design procedure. It is traditionally performed manually, i.e. via a trial and error procedure. The objective is to shape the nominal open-loop frequency response of the system so that:

- It lies above the robust-performance contours,
- It avoids the robust stability region (\mathcal{U} -contour), and
- It encircles the critical point in the anti-clockwise direction p times, where p is the number of open-loop unstable poles.

It is implicitly assumed that the whole family of uncertain plants has the same number of unstable poles, and that

no unstable pole/zero cancellations occur between the feedback controller and $G(p, s)$ for every $p \in \mathcal{P}$. Although this last condition requires independent verification, it is rarely an issue in practice designs since only stable and minimum-phase controllers are typically employed.

Among all possible controllers which meet the above requirements, the "best" design is considered to be the one in which the open-loop frequency response at the design frequencies lies as close as possible to the robust performance templates. This is in order to avoid "over-designing" the system by using excessively large gains, which may lead to noise amplification, instability due to unmodelled dynamics, etc. Very sensibly, the QFT design philosophy demands the use of "as little feedback as possible" consistent with the robust performance specifications.

The software provides an interactive design tool to help with manual loop-shaping and three optimisation algorithms for automatically designing "optimal" controllers which meet the QFT constraints described above. These are:

- An algorithm for designing optimal fixed-structure controllers (PID, first-order lead/lag, second-order with complex poles),
- An algorithm for designing controllers in the frequency-domain via linear programming, and
- An convex optimisation algorithm for designing the optimal controller indirectly by shaping the closed-loop transfer function

The manual loop-shaping tool and the three optimisation algorithms are described in more detail in the following four sections.

2.3.1 Manual loop-shaping tool

This is an interactive graphical design tool for shaping the open-loop characteristics of the system according to the QFT constraints described above. By clicking on two points on the Nichol's chart, the required frequency and phase/gain differences are calculated and an appropriate first-order phase lead/lag controller is automatically designed. Alternative options include the addition of integral, derivative, proportional or second-order terms with complex roots. The uncompensated and compensated frequency responses are next displayed on the Nichol's chart. Based on this graphical output the designer can proceed with the design by adding a new factor in the compensator, back-step by deleting the last compensator factor designed, etc. The overall logic of the tool summarising the options available to the designer at each stage are shown in the flow-diagram of Figure 3. Note that usually the initial controller $K_{\text{arg}} = 1$.

Clearly, there is not a unique way for designing a compensator using this method, and often a successful design requires a considerable amount of experience. When the specifications are tight, the design procedure may require a large number of cascade terms, resulting in a high-order overall controller. To help with this potential

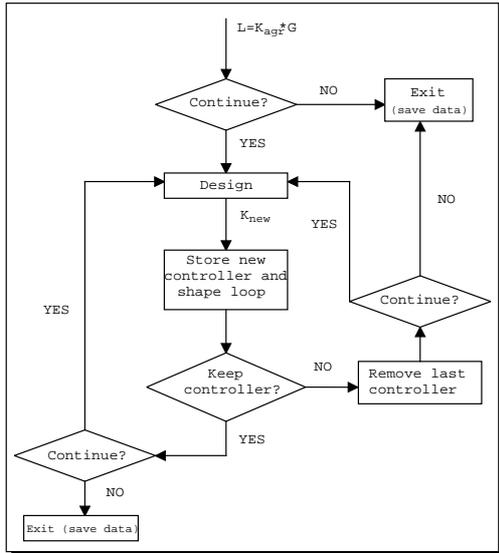


Figure 3: Lead/Lag-network manual loop shaping tool

rise in complexity, a controller model-reduction tool is supplied. The effectiveness of this reduction procedure may be checked by displaying the frequency responses of the designed (high-order) system and its low-order approximation on the Nichol's chart. The tool was used

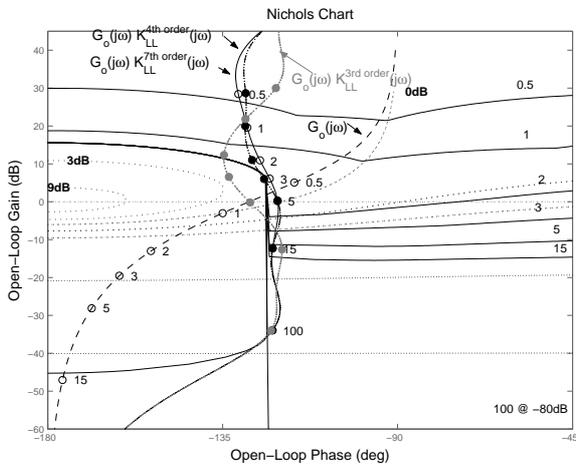


Figure 4: Control Design with Lead/Lag Network,

to design a feedback controller for the system introduced in the first paragraph of section 2. The resulting frequency-response of open-loop system (with the designed 7-th order controller (shown as -o-)), along with the nominal plant are displayed in Figure 4. It may be seen that all robust stability and performance specifications are satisfied; in addition the open-loop frequency response points at the seven design frequencies are reasonably close to the corresponding robust performance templates. When controller approximation was performed (via the Schur balanced model-reduction method) it was found that up to

3 states could be removed without affecting significantly the controller's frequency response. The corresponding nominal open-loop response using a 4th-order (shown as -o-) and a 3rd-order controller are also shown in Figure 4 for comparison.

2.3.2 Fixed-structure controller optimisation

The algorithms described in this section are implementations of the methods in [12] and [10]. The optimisation is carried out over the parameters of fixed-structure controllers (PID, first-order lead/lag, second-order). The main idea behind the algorithm is that, for selected control structures, fixing the phase of the controller at *two* distinct frequencies, determines completely the phase response of the controller over *all* frequencies and thus also the phase of the nominal open-loop system. Thus, using the robust stability and performance bounds on the Nichol's chart, it is straightforward to determine the minimum controller gain (if it exists) so that all robust stability and performance objectives are satisfied. (Note that if the phase of the nominal open-loop system is fixed, varying its gain corresponds to shifting the response vertically in the Nichol's chart). Repeating the procedure over all phase combinations (suitably discretised) will produce the optimal controller parameters. Optimality in this context may be defined in terms of asymptotic open-loop gain, nominal/worst-case cross-over frequency or closed-loop bandwidth, or some other appropriate measure consistent with the general QFT philosophy penalising system "over-design".

The numerical techniques used to implement the algorithm include: (a) Singular value decomposition (for phase-lead/lag compensation only), and (b) A robust stability gain margin calculation. This can be performed via either a numerical algorithm (when the nominal transfer function is known) or via purely graphical means (i.e. directly from the frequency response) by counting the crossings on a certain line on the Nichol's chart and its directions (see [4], [3] for details).

The nominal open-loop frequency response corresponding to the optimal PID controller for the example in section 2 is shown in Figure 5 below. Note that the gain for all design frequencies lies above the corresponding robust performance bounds, with one lying exactly on the robust stability boundary (\mathcal{U} -contour). The asymptotic phase of the loop is -90° since the controller includes a pure derivative term. A simple modification to the algorithm is required if one wishes to limit the derivative action at high frequencies (see [12] for details).

The algorithms described in this section may be used to design simple controllers or as a first step in a more complex design. Note that every rational controller of arbitrary complexity can be constructed from cascade interconnections of the types used here (i.e. integrator, phase lead/lag, second-order denominator or numerator term). Thus, it is possible to improve the design continuously by building high-order controllers in a step-by-step procedure: At each step the optimisation algorithm is carried out (for one of the above three struc-

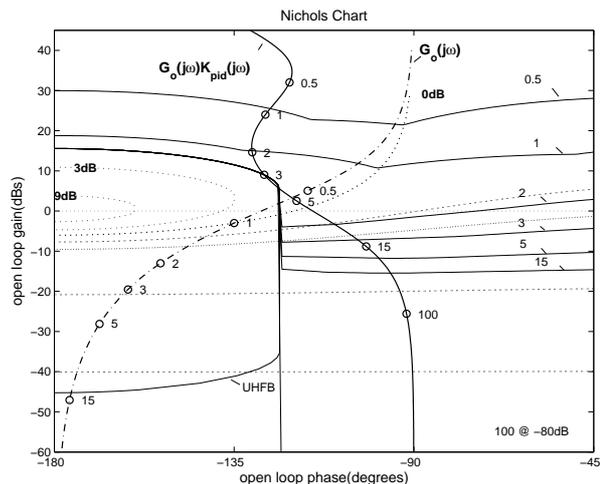


Figure 5: Design with Optimal PID controller

tures) and the resulting optimal controller $K(s)$ is accumulated into the nominal open-loop system by redefining $L_o(s) \leftarrow L_o(s)K(s)$. This process may continue until a satisfactory design is obtained, or until the “cost” fails to decrease significantly. Of course, the controller resulting from this procedure will not, in general, be optimal over the higher-order controller set!

2.3.3 Frequency-domain controller optimisation via linear programming

The algorithms described in this section is based on [1]. The optimisation is carried out directly over the frequency response characteristics of the controller (equivalently the nominal open-loop) at the design frequencies. The robust stability and performance bounds (restricted to appropriate phase-ranges) are approximated by linear inequalities and define the constraints of the resulting Linear Programming optimisation problem. The objective function which is minimised involves the asymptotic open-loop gain or a weighted average of the controller gains over all design frequencies. To ensure that the optimised frequency response can be realised by a stable and minimum-phase transfer function, Bode’s gain-phase integral relationship is discretised over a suitable frequency grid and is augmented to the constraints of the Linear Programme (in the form of linear equality constraints). The transfer function of the controller is finally fitted from the optimised frequency-response data via a weighted least-squares approximation technique. A practical design example which employs this method will be presented at the Conference.

2.3.4 Loop shaping via closed-loop convex programming

This algorithm, based on [2], relies on algebraic factorisation theory and convex optimisation techniques. Rather than optimising the open-loop frequency response, the

QFT problem is posed directly in terms of the complementary sensitivity function T , at a discrete set of design frequencies. Closed-loop stability is imposed via “interpolation constraints”, involving the value of the complementary function (and its derivatives) at every unstable pole and non-minimum phase zero of the nominal plant. The robust performance specifications (involving disturbance-rejection only at this stage) are formulated as regions in the complex plane in which $T(j\omega_i)$ must lie at each design frequency ω_i . Since these are not convex, in general, they are approximated by an appropriate convex polygon or ellipsoid. A convex optimisation problem is finally formulated and solved in terms of the free numerator parameters of $T(s)$, and the optimal controller is recovered from the optimal $T(s)$. The main draw-back of this approach is that the closed-loop poles are fixed at arbitrary locations via a stable polynomial to make the optimisation problem tractable. However, a careful selection of these poles often produces good QFT designs which may be further improved via open-loop shaping. Alternatively, the closed-loop poles corresponding to an initial design may be used in the optimisation to improve it further.

2.4 Design of pre-filter

This is a simple exercise in filter design which may be performed manually or via an optimisation-based tool which fits the magnitude frequency response of the filter at a discrete set of frequencies. A pre-filter is only necessary when the design includes robust tracking objectives. Provided the QFT specifications have been met, the worst-case spread in closed-loop gains $|\Delta T(p, j\omega_i)|$ is guaranteed to be less than $\delta(\omega_i)$ for each design frequency ω_i . Thus, the purpose of the pre-filter is to scale the closed-loop gains so that they all lie within the specified bounds $B_l(\omega)$ and $B_u(\omega)$.

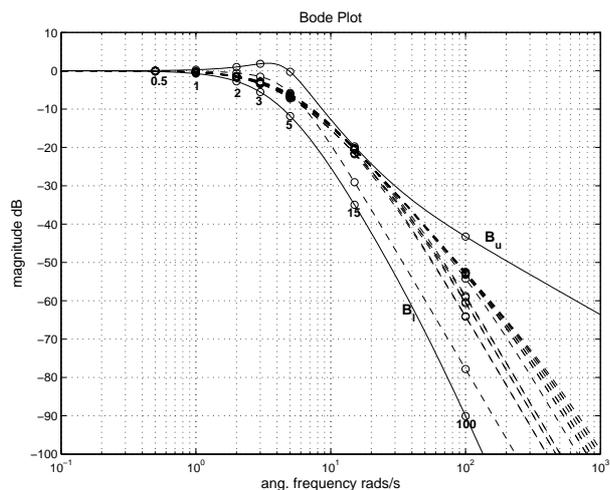


Figure 6: Closed Loop Frequency Response and Tracking Bounds

Figure 6 shows the shaped closed-loop frequency responses $FGK(1 + GK)^{-1}(s)$ for the example in of sec-

tion 2, with $K(s)$ the optimal PID controller of section 2.3.2. The designed pre-filter is a second-order filter given by

$$F(s) = \frac{1}{\left(\frac{s}{3.5} + 1\right)\left(\frac{s}{7.0} + 1\right)}$$

A wide range of parameter pairs (a, k) have been selected for the plot, including the four extreme combinations $(1, 1)$, $(1, 10)$, $(10, 1)$ and $(10, 10)$. As expected, all responses lie in the region between the two bounds, which verifies that the design has been successful.

5. CONCLUSIONS

The paper has presented the main features of a new CAD tool for robust control design using the QFT method. The software may be used as an educational tool for introducing the topics of QFT and robust control via small/medium-size design problems and as a test-bed for developing new techniques in this area. Most algorithms presented in the paper have been implemented and tested. It is hoped that a full working version of the software will be ready by the end of the year and will become available to the academic community via the net.

REFERENCES

- [1] G. F. Bryant and G. D. Halikias. Optimal loop-shaping for systems with large parameter uncertainty via linear programming. *International Journal of Control*, 63(3):557–568, 1995.
- [2] Y. Chait, Q. Chen, and C.V. Hollot. Automatic loop shaping of QFT controllers via linear programming. *preprint*.
- [3] W. Chen and D. J. Ballance. Stability analysis on the Nichols chart and its application in QFT. *Preprint, University of Glasgow*, August 1997.
- [4] W. Chen and D. J. Ballance. QFT design for uncertain non-minimum phase and unstable plants. *Preprint, University of Glasgow*, July 1997.
- [5] J. D’Azzo and C. Houpis. *Feedback control systems analysis and synthesis*. Prentice Hall, 1998.
- [6] I. M. Horowitz and M. Sidi. Synthesis of linear systems with large plant ignorance for prescribed time-domain tolerances. *Int. J. Control*, 16:287–309, 1972.
- [7] I. M. Horowitz and M. Sidi. Optimum synthesis of non-minimum phase systems with plant uncertainty. *Int. J. Control*, 27:361–386, 1978.
- [8] I. M. Horowitz. *Synthesis of Linear Systems*. Academic Press, 1973.
- [9] C. Houpis and Rasmussen S.J. *Quantitative Feedback Theory Fundamentals and Applications*. Marcel Dekker, 1999.
- [10] R. Nandakumar, G.D. Halikias, and A. Zolotas. An optimisation algorithm for designing fixed-structure controllers using the QFT method. *IEEE CCA02-CACSD*, September 2002.
- [11] O. Yaniv and I. M. Horowitz. A quantitative design method for MIMO linear feedback systems having uncertain plants. *Int. J. Control*, 43:401–421, 1986.
- [12] A. C. Zolotas and G. D. Halikias. Optimal design of PID controllers using the QFT method. *IEE Proceedings - Control Theory Appl.*, 146(6), November 1999.