

QFT CONTROL TOOLBOX (QFTCT)

USER'S GUIDE

F-1 INTRODUCTION

This appendix presents the interactive object-oriented CAD tool for QFT controller design included with the book: the QFT Control Toolbox (QFTCT), ver.1.01. It includes the latest technical quantitative robust control achievements included in the book within a user-friendly and interactive environment.

The toolbox has been developed by Professor Mario Garcia-Sanz and Augusto Mauch at the Public University of Navarra (UPNA) and Case Western Reserve University (CWRU), and by Dr. Christian Philippe at the European Space Agency (ESA-ESTEC). It was also supported by MCyT (Ministerio de Ciencia y Tecnología, Spain).

See the web page of the *Control and Energy Systems Center*, at Case Western Reserve University, for details. <http://cesc.case.edu>

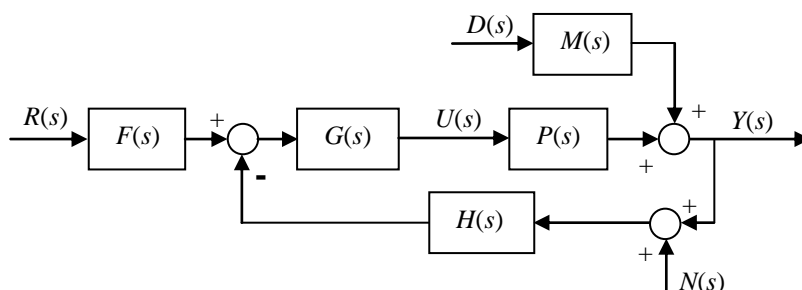


Fig. F.1 Multi-Input-Single-Output feedback control system.

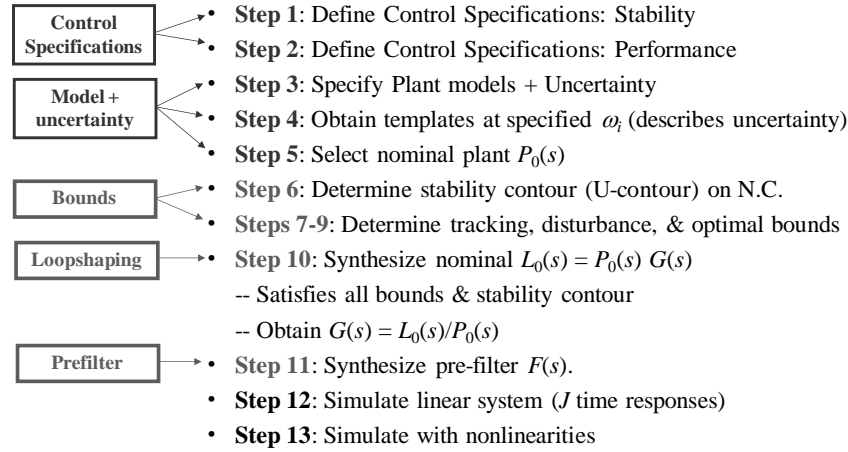


Fig. F.2 QFT design procedure

The QFT Control Toolbox (QFTCT) runs under MATLAB[®] and shows a special architecture based on seven principal windows (W1, *Plant Definition*; W2, *Templates*; W3, *Control Specifications*; W4, *Bounds*; W5, *Controller Design*; W6, *Pre-filter Design*; W7, *Analysis*) and a common central memory.

It also includes a library of basic and advanced functions to be selected in the corresponding windows, allows a multitasking/threading operating system, offers a user-friendly and interactive environment by using an object-oriented programming, permits easily to rescale the problem from single-input-single-output (SISO) to multiple-input-multiple-output (MIMO), and uses reusable code.

The QFT Control Toolbox CAD package includes the latest quantitative robust control system design achievements within a user-friendly and interactive environment. The main objective is to design and implement a 2DOF (two degree of freedom) robust control system, see Fig. F.1, for a plant with uncertainty, which satisfies the desired performance specifications, while achieving reasonably low loop gains (avoiding or minimizing bandwidth, sensor noise amplification and control signal saturation).

The compensator $G(s)$ and a pre-filter $F(s)$ are to be designed to meet robust stability and robust performance specifications, and to deal with reference tracking $R(s)$, disturbance rejection $D(s)$, signal noise attenuation $N(s)$ and control effort minimization $U(s)$, while reducing the cost of the feedback (excessive bandwidth).

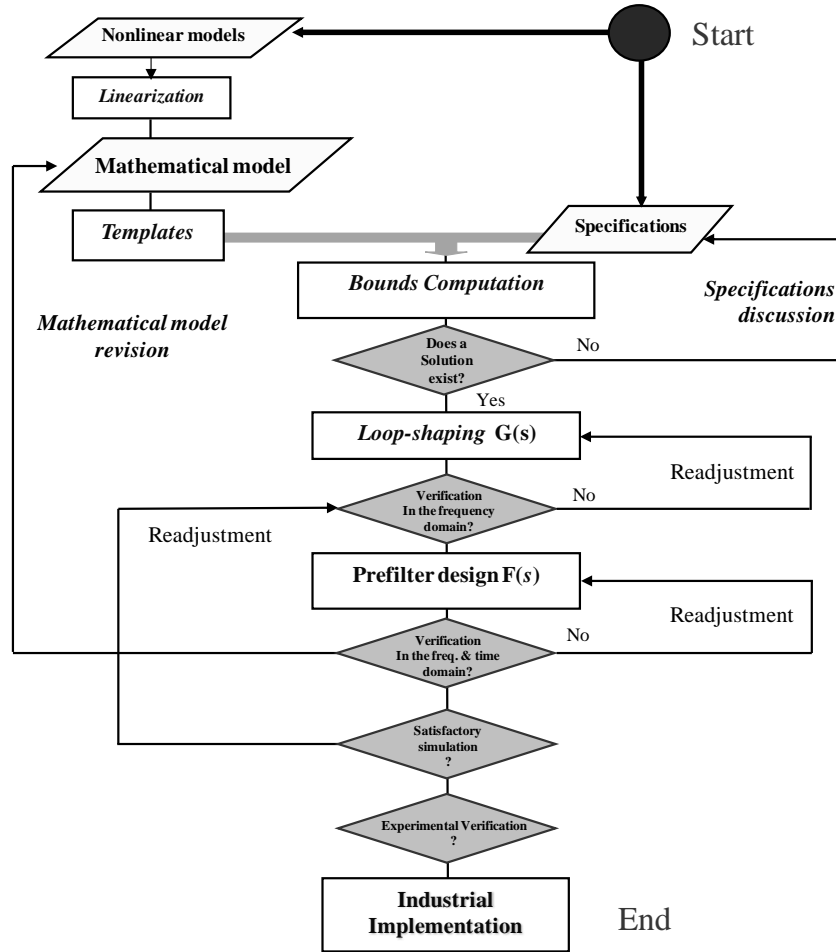


Fig.F.3 QFT flowchart for a MISO control system.

Fig. F.2 lists the steps involved in the QFT design procedure. Fig. F.3 represents the CAD flowchart of the MISO analog QFT design procedure, and Fig. F.4 presents an overview of the QFT design process.

The QFTCT files, user's manual and examples can be free downloaded from the websites <http://cesc.case.edu> and <http://www.crcpress.com>

F-2 QFT CONTROL TOOLBOX WINDOWS

This section presents an overview of the QFT design procedure for this QFT Control Toolbox (QFTCT) CAD Package.

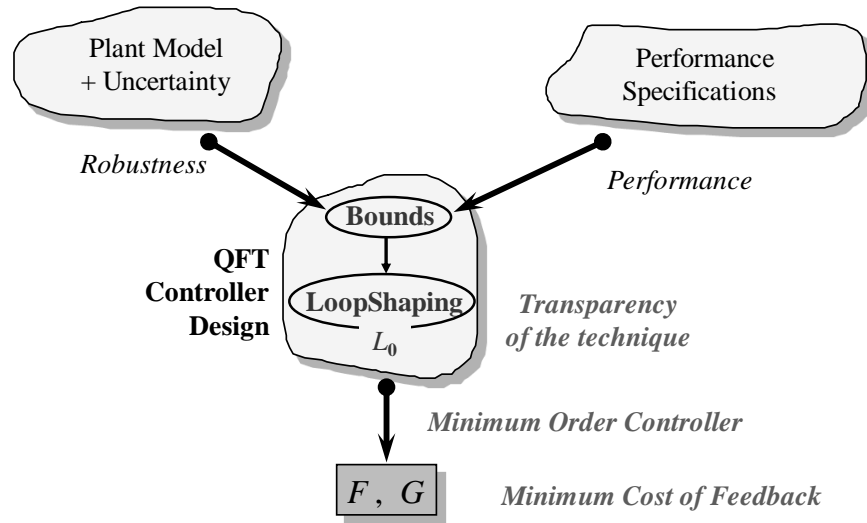


Fig. F.4 An overview of the QFT design process.

Required products: Matlab 7.9.0 (R2009b) or later version;
Control System Toolbox 8.4 or later version.

Platform: PC or Mac.

Installation: To install the QFTCT, copy the folder containing the toolbox to a safe place (it can be any folder) and then add that folder to Matlab paths (file → set path → add folder → save → close).

Start: To start a new session with the QFT Control Toolbox, type the following command in the Matlab Command Window:
QFTCT.

F-2.1 GENERAL DESCRIPTION

- **Windows** The toolbox contains windows of the form shown in Fig. F.5. They fall into seven categories: W1, *Plant Definition*; W2, *Templates*; W3, *Control*

Specifications; W4, *Bounds*; W5, *Controller Design*; W6, *Pre-filter Design*; W7, *Analysis*. The windows have a “bar” at the top, listing these seven categories as shown in Fig. F.6.

All the windows have a navigator panel (see Fig. F.5) in the top that allows the user to change the active window. Not all the windows are available from the beginning, i.e. some windows have pre-requisites that have to be fulfilled in order to activate the windows. The seven categories are listed next, following the order that they have to be executed in the design process:

1. Plant definition: No pre-requisites
2. Templates: Needs Plant definition
3. Specifications: Needs Plant definition
4. Bounds: Needs Plant definition, Templates and Specifications
5. Controller Design: Needs Plant definition, Templates, Specifications and Bounds
6. Pre-filter: Needs Plant definition, Templates, Specifications, Bounds and Controller design. It is only for Reference tracking problems
7. Analysis: Needs Plant definition, Templates, Specifications, Bounds and Controller design.

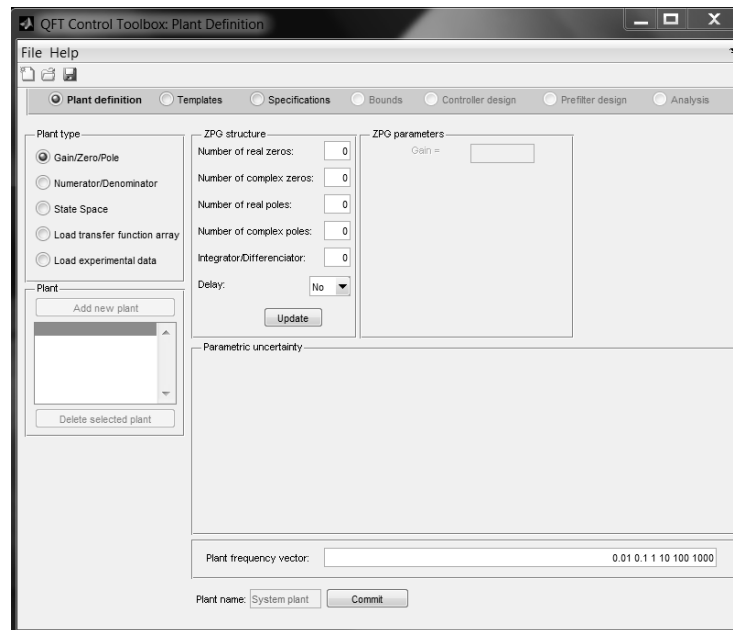


Fig. F.5 Plant definition window.

Mario Garcia-Sanz, Augusto Mauch, Christian Philippe, et al. (2011). *The QFT Control Toolbox (QFTCT) for MATLAB*. Case Western Reserve University. <http://cesc.case.edu>

Mario Garcia-Sanz, Constantine H. Houpis, (2012), *Wind Energy Systems: Control Engineering Design*, CRC Press, Taylor & Francis. ISBN: 978-1-4398-2179-4. (Appendix F: *QFT Control Toolbox User's Guide*).

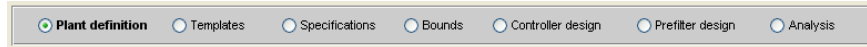


Fig. F.6 Window toolbar, listing the seven window categories.



Fig. F.7 Toolbars for windows: (a) W1 and W3, (b) W2, W4, W5, W6 and W7.

- **Toolbar:** All the windows of the toolbox have a toolbar [see Fig. F.7(a)], which allows the user to create a new project, open an existing one and save the current project. The toolbar of the windows which have plots (Templates, Bounds, Controller Design, Pre-filter Design and Analysis) has controls to zoom in and out on the plots [see Fig. F.7(b)].
- **Menu** All the windows have the following two menus:
 - **File:** Allows the user to create new projects, open existing projects, save the current project and save the current project in a new file.
 - **Help:** Provides access to the User's Guide and to the information about the toolbox (authors, version, etc).

In addition,

- The windows which have plots have a menu item in the File menu that allows saving the plot in a .emf, .bmp and .fig file.
- Some window categories have specific menu items, for example:
 - **Templates.** In the *File menu*, there is a menu item that allows the user to export the nominal plant as a transfer function.
 - **Controller Design.** In the *File menu* there are menu items that allows the user to (see Fig. F.8):
 - ⊙ Load a controller from the hard drive (*.contr)
 - ⊙ Load a list of controllers from the hard drive (*.contrList)
 - ⊙ Save selected controller to the hard drive (*.contr).
 - ⊙ Save controller list to the hard drive (*.contrList)
 - ⊙ Export the selected controller as a transfer function to workspace.
 - ⊙ Save Figure to the hard drive (*.emf, or *.bmp, or *.fig)

- ⊙ *Check the stability.* The matlab *isstable* function was used to verify the stability of the closed loop system with the selected controller.

The controller design window has also an Edit menu that allows the user to:

- ⊙ *Undo changes*
- ⊙ *Redo changes*
- ⊙ *Set the frequency vector for $L(s)$*

▪ **Pre-filter Design** Similarly, this window category has the following menu items in the *File menu*:

- ⊙ *Load pre-filter* from the hard drive (*.prefltr)
- ⊙ *Load pre-filter list* from the hard drive (*.prefltrList)
- ⊙ *Save selected pre-filter* to the hard drive as (*.prefltr).
- ⊙ *Save pre-filter list* to the hard drive (*.prefltrList)
- ⊙ *Export* the selected pre-filter as a transfer function to the workspace.
- ⊙ *Save Figure* to the hard drive (*.emf, or *.bmp, or *.fig)

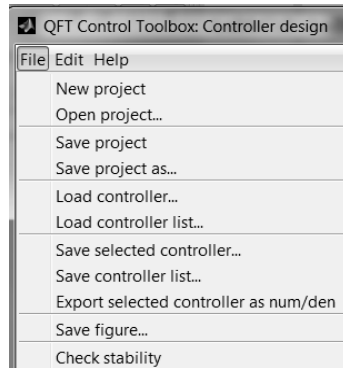


Fig. F.8 File Menu, Controller design window.

F-2.2 PLANT DEFINITION WINDOW

In the Plant Definition Window (W1), see Fig. F.9, the user can define the plant model structure, the parameters, the uncertainty and the frequencies of interest. There are several panels in this window. In the first one, Plant type, the user can select the way to describe the plant, which can be: (1) Gain/Zero/Pole transfer functions, (2) Numerator/Denominator transfer functions, (3) State Space

representation, (4) Multi-structure transfer function arrays, (5) Experimental data.

- **Gain/Zero/Pole Transfer Function.** The model structure and its elements are defined using the syntax listed in Table F.1 and Eq. (F.1). In the first step the user has to enter the structure of the plant: number of zeros, complex zeros, poles and complex poles (see Figs. F.9 and F.10). The user also has to enter the value of the integrator/differentiator element (0 if it is not used, a positive integer for differentiators and a negative one for integrators) and to specify if the plant has time-delay. After this is accomplished, the user has to press the “Update” button. Then the Toolbox updates the second ZPG panel, and the user can enter the expressions for the elements of the plant.

Note that, at this point, the user can introduce numbers or letters. If the user introduces letters, the Toolbox identifies them as parameters with uncertainty, and automatically adds their names to the Parametric uncertainty panel, which will be define afterwards. In Figs. F.9 and F.10 a gain is introduced as “ $k/(ab)$ ”, a real pole as “ a ”, and another real pole as “ b ” [see Eq. (F.2) as well].

QFT Control Toolbox: Plant Definition

File Help

Plant definition Templates Specifications Bounds Controller design Prefilter design Analysis

Plant type

- ☒ Gain/Zero/Pole
- ☐ Numerator/Denominator
- ☐ State Space
- ☐ Load transfer function array
- ☐ Load experimental data

Plant

Add new plant

System plant

Delete selected plant

ZPG structure

Number of real zeros: 0

Number of complex zeros: 0

Number of real poles: 2

Number of complex poles: 0

Integrator/Differentiator: -1

Delay: No

Update

ZPG parameters

Gain = $k/(a*b)$

Real pole 1 = a

Real pole 2 = b

Continue

Parametric uncertainty

Name	Uncertainty/Probability	Min	Max	Grid	Nominal
a	Modify	1	15	S log	1
b	Modify	150	170	S log	150
k	Modify	610	1050	S log	610

Plant frequency vector: 0.01 0.1 0.5 1 2 5

Plant name: System plant Update

Fig. F.9 Plant definition window.

$$P(s) = \frac{k \text{ zeros}(s)}{\text{poles}(s)} = \frac{k \left(\frac{s}{z_1} + 1 \right) \left(\frac{s}{z_2} + 1 \right) \cdots \left(\frac{s^2}{\omega_{ni}^2} + \frac{2\zeta_i s}{\omega_{ni}} + 1 \right) \cdots}{s^r \left(\frac{s}{p_1} + 1 \right) \left(\frac{s}{p_2} + 1 \right) \cdots \left(\frac{s^2}{\omega_{nj}^2} + \frac{2\zeta_j s}{\omega_{nj}} + 1 \right) \cdots} e^{-sT} \quad (\text{F.1})$$

Fig. F.10 ZPG structure and parameters panels.

$$P(s) = \frac{k / (ab)}{s \left(\frac{s}{a} + 1 \right) \left(\frac{s}{b} + 1 \right)} = \frac{k}{s(s+a)(s+b)} \quad (\text{F.2})$$

Table F.1 Gain/Zero/Pole/Delay element syntax.

Gain	k	Complex zero	$(s / \omega_n)^2 + (2\zeta / \omega_n)s + 1$ ($\zeta < 1$)
Real pole	$\frac{1}{(s / p) + 1}$	Integrator	$\frac{1}{s^n}$
Real zero	$(s / z) + 1$	Differentiator	s^n
Complex pole	$\frac{1}{(s / \omega_n)^2 + (2\zeta / \omega_n)s + 1}$ ($\zeta < 1$)	Time delay	e^{-Ts}

- **Numerator/Denominator Transfer Function:** The plants can also be defined as a transfer function, with numerator and denominator Laplace polynomials [see Eq.(F.3)]. The first step consists in entering the model structure: numerator and denominator polynomial orders [see Fig F.11 and

Eq. (F.3)]. At this point the user can also enter the delay of the plant. After pressing the Update button, the Toolbox updates the Numerator and Denominator panels, and the user can enter the expressions for the coefficients of the polynomials and delay. Note that, at this point, the user can introduce numbers or letters. If the user introduces letters, the Toolbox identifies them as parameters with uncertainty, and automatically adds their names to the parametric uncertainty panel, which will be defined afterwards. In Fig. F.11, the expression defined in Eq.(F.4) is introduced.

$$P(s) = \frac{n(s)}{d(s)} = \frac{Ncoef_n s^n + Ncoef_{n-1} s^{n-1} + \dots + Ncoef_1 s + Ncoef_{indep} e^{-sT}}{Dcoef_m s^m + Dcoef_{m-1} s^{m-1} + \dots + Dcoef_1 s + Dcoef_{indep}} \quad (\text{F.3})$$

Fig. F.11 Numerator/denominator panels.

$$P(s) = \frac{Lambda}{(M_1 M_2) s^4 + 0 s^3 + Lambda (M_1 + M_2) s^2 + 0 s + 0} \quad (\text{F.4})$$

- **State Space** The plants can also be described by using a state space representation. In the first step the user has to enter the structure or dimensions of the matrixes A ($n \times n$), B ($n \times m$), C ($l \times n$) and D ($l \times m$) [see Eq.(F.5) and Fig. F.12]. After pressing the Update button, the Toolbox updates the State Space matrices panel, and the user can enter the expressions for the elements of the four matrices. Note that, at this point, the user can introduce numbers or letters. If the user introduces letters, the Toolbox identifies them as parameters with uncertainty, and automatically adds their names to the parametric uncertainty panel, which is defined later.

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u} \\ \mathbf{y} &= \mathbf{C} \mathbf{x} + \mathbf{D} \mathbf{u} \end{aligned} \quad (\text{F.5})$$

$\begin{matrix} n \times 1 & n \times n & n \times m \\ l \times 1 & l \times n & l \times m \end{matrix}$

Fig F.12 State Space panels.

- **Load Transfer Function Array** If the plant cannot be defined using the three previous structures, the user can also load an array of transfer functions from the hard drive. After uploading the array, the user has to enter the row of the array that represents the nominal plant. This is a very powerful tool that can define any kind of plant with different structures and parametric and non-parametric uncertainties.

For the *SystemPlant* the technique is as follows: (1) Run first in Matlab a m.file like the one describe next; (2) Then go to the Workspace; (3) Click on “P” with right bottom and Save as PP.mat (or other name) in the hard drive; (4) Then go to the Plant definition window; (5) Click “Load transfer function array”; (6) Select Nominal plant (usually num.1); (7) Click “Import.mat”; (8) Select in the hard drive PP.mat and click “Open”; (9) Click “Commit”; (10) The plant will appear in the list of plants as *System Plant*.

For an additional plant: steps (1) to (6) are the same –now save as D.mat, for instance. Then (7) Click “Add new plant”; (8) Put a name in the “Plant name” cell at the bottom (for example DD); (9) Click “Import.mat”; (10) Select in the hard drive D.mat and click “Open”; (11) Click “Commit”; (12) The plant will appear in the list of plants as *DD*.

Example:

```
c = 1;
for k=linspace(610,1050,3)
    for a=linspace(1,15,15)
        for b=linspace(150,170,2)
            P(1,1,c)=tf(k,[1 (a+b) a*b 0]);
            c=c+1;
        end
    end
end
```

Note that we have three nested “for” loops in the above example because there are three parameters with uncertainty: $k \in [610, 1050]$, $a \in [1, 15]$, $b \in [150, 170]$. We can also put another kind of grid, different from *linspace*, like *logspace* or others, or a mix. The definition of the plant in this example is:

“ $P(1,1,c)=tf(k,[1 (a+b) a*b 0])$ ”,
 which is: $P(s) = k/[s^3+(a+b)s^2+ab s]$

Note also that this is a very powerful option that allows the user to include in this line of the algorithm any kind of structure or expression, even many different structures (uncertainty in the structure) with parametric and non-parametric uncertainty, interdependence in the uncertain parameters and many other special requirements.

- **Load Experimental Data** The user can also define the system by uploading experimental data in a frequency-response-data vector. The technique is as follows: (1) Prepare a frd (Frequency Response Data model) system in Matlab: *freq* = vector of frequencies in rad/sec; *resp* = vector of complex numbers with the response of the system at each frequency ($a+jb$), *experimentalData* = frd(*resp*,*freq*). Note that the name of the frd structure can be *experimentalData* or any other.

For example, in Matlab: *freq* = logspace(1,2); *resp* = 0.05*(*freq*).*exp(i*2**freq*); *experimentalData* = frd(*resp*,*freq*), or a collection of real data “*resp*”; (2) Then go to the Workspace; (3) Click on “*experimentalData*” with right button and save as *sys.mat* (or other name) in the hard drive; (4) Then go to the Plant definition window; (5) Click “Load experimental data”; (6) Select Nominal plant (usually num.1); (7) Click “Import experimental...”; (8) Select in the hard drive *sys.mat* and click “Open”; (9) Click “Commit”; (10) The plant will appear in the list of plants as *System Plant*.

- **Expressions and Parametric Uncertainty** As seen, it is possible to introduce alpha-numeric expressions. The toolbox automatically recognizes letters as parameters with uncertainty. After entering the expressions of the plant (the parameters in the zpk model description, the coefficients in the numerator/denominator model description, etc) and pressing the “Continue” button, a panel appears where the user is able to define the parametric uncertainty of the plant (see Fig. F.13).

This panel displays the following information of each parameter:

Parametric uncertainty					
Name	Uncertainty/Probability	Min	Max	Grid	Nominal
a	<input type="button" value="Modify"/>	1	15	<input type="text" value="5"/> <input type="button" value="log"/>	<input type="text" value="1"/>
b	<input type="button" value="Modify"/>	150	170	<input type="text" value="5"/> <input type="button" value="log"/>	<input type="text" value="150"/>
k	<input type="button" value="Modify"/>	610	1050	<input type="text" value="5"/> <input type="button" value="log"/>	<input type="text" value="610"/>

Fig. F.13 Parametric uncertainty panel.

- The name of the parameter.
 - A button that will open a window to modify the probability distribution of the parameter.
 - The minimum and the maximum value of the parameter (given by its probability distribution).
 - Number of points in the grid for each parameter (two is the minimum number, which means that there are only two points in the grid, the minimum and the maximum), and its distribution (logarithmic or lineal).
 - The nominal value of the parameter (must be included in the minimum - maximum range of the parameter).
- **Probability distribution of the parameters** In order to define the probability distribution for the uncertainty of the parameters, the user needs to press the “Modify” button in the “Parametric uncertainty” panel and an emerging window will appear (see Fig. F.14). There are three probability distributions available: Uniform, Normal and Weibull. After selecting the distribution and entering its parameters, a graphical representation of the distribution is plotted. After that, the user can define the percentage to be reached. This percentage is applied to the distribution, and the resulting values will be the minimum and the maximum values of the parameter.

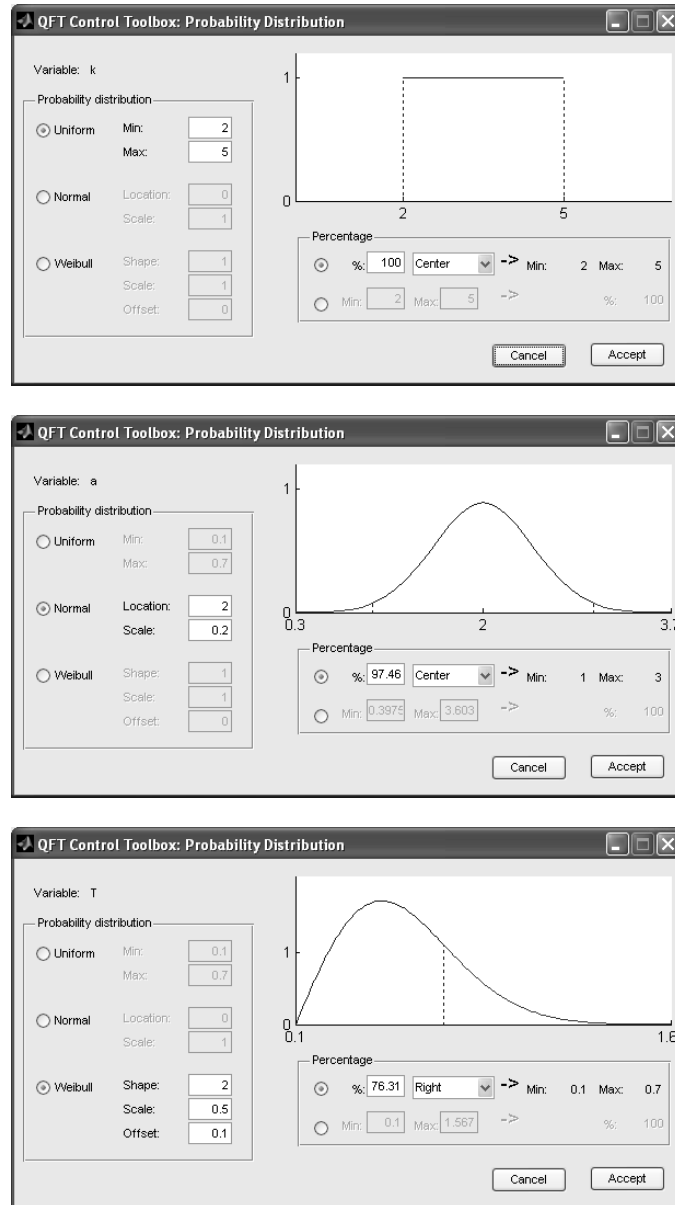


Fig. F.14 Probability distribution: Uniform, Normal and Weibull cases.

Mario Garcia-Sanz, Augusto Mauch, Christian Philippe, et al. (2011). *The QFT Control Toolbox (QFTCT) for MATLAB*. Case Western Reserve University. <http://cesc.case.edu>

Mario Garcia-Sanz, Constantine H. Houppis, (2012), *Wind Energy Systems: Control Engineering Design*, CRC Press, Taylor & Francis. ISBN: 978-1-4398-2179-4. (Appendix F: *QFT Control Toolbox User's Guide*).

- **System plant frequency vector.** The user has to enter a vector with the frequencies of interest of the plant (see for example Figs. F.5 or F.9). The textbox accepts a number array or a Matlab command that produces a number array [for example, `logspace (-2,3,100)`]. Note that it is very important to define well this vector, populating the vector with an enough number of points in the frequency regions where there are resonances or quick changes in magnitude or phase, as well as selecting properly the lowest and the highest value of frequency.

F-2.3 TEMPLATES WINDOW

A template is the representation of the frequency response of the plants, including the uncertainty, in the Nichols Chart at a particular frequency. There is a specific window to define the templates (see Fig. F.15).

- **Number of Template Points:** The number of points of the templates depends on three parameters: template type, parametric uncertainty and template contour. If the template is too sparse, then it may be not accurate enough. If too few points are used, the computed bounds are not relevant to the original plant description whose boundary is a continuous smooth curve.
 - **Template type** The user can select four different template types, ordered from sparse to dense: “Vertex”, “Edges”, “Faces” and “All points”. Fig. F.16 shows different template types for a plant which has three uncertain parameters (the grid of two of them is 5 points, and the grid of the remaining one is 10 points). The filled circle in each of the templates types denotes the template's nominal point.
 - **Grid of the Parametric Uncertainty Variables** The more points are sampled (Fig. F.13), the denser the templates are (Note that if the Template-type is “Vertex,” this grid does not affect the template's figure).
 - **Template Contour** In the lower part of the window there is a slider which can be used to adjust the contour of the templates. If the slider is moved to the right the *inner points* of the template disappear.

But if the slider is moved to the right too much then the *points of the contour* become sparser too, and the contour is not representative. Fig. F.17 shows the template of a plant which has initially 1000 points (a), and the template of the same plant after its contour has been adjusted (b).

- **Plant Frequencies** There is a template for each plant frequency. The user can add and remove frequencies using the templates window.

- **Add Frequency** Frequencies can be added using two different methods:
 - By entering the value of the new frequency in the textbox beside “Manual” and pressing the “Add button” (see Fig. F.15).
 - By using the slider (see the frequency panel in Fig. F.15). When the user clicks the “Initialize button” in the “Add frequency panel”, the slider is enabled and the user can pre-visualize the position and shape of the new template. The user can then use the slider to enter the value of the new frequency. Then, clicking the “Fix” button, the new frequency is added to the list.

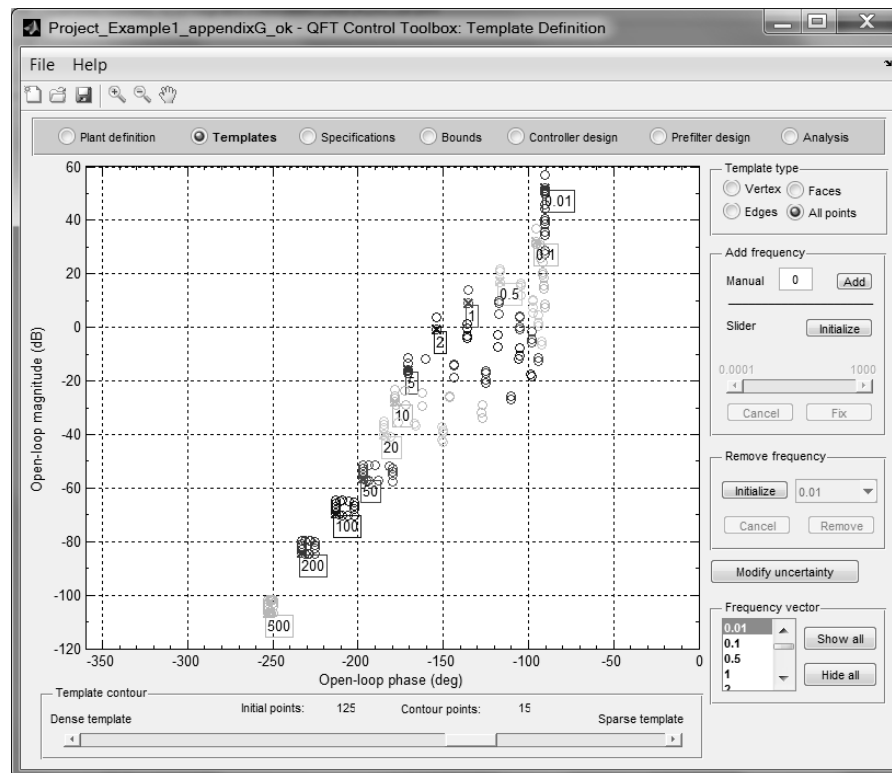


Fig. F.15 Template definition window.

- **Remove Frequency** To remove a frequency, the user has to click the “Initialize” button in the “Remove frequency panel”. A list box appears, and enables the user to select the frequency to be removed. The points

markers of the template associated with the frequency are indicated by 'x' instead of 'o'. To remove the selected frequency, click "Remove".

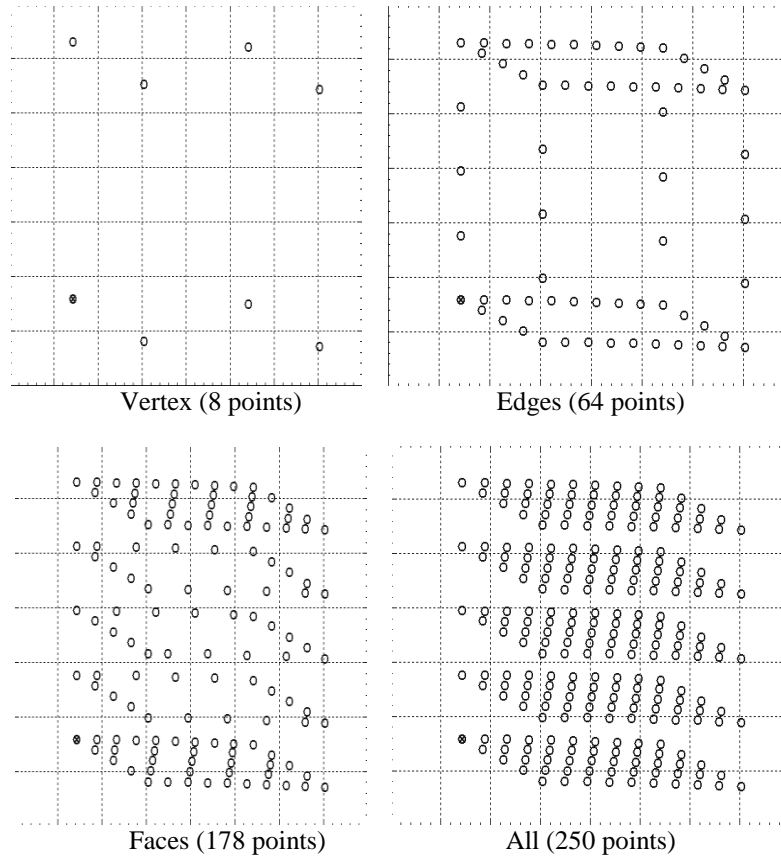


Fig. 16 Templates Types.

- **Frequency Vector** The user can use the "Frequency vector" panel (see Fig. F.15) to change the visibility of the templates associated with the frequencies. If the user double clicks on a number of the frequency list that represents a template, the visibility of the template is switched (on/off). The user can use the "Show all" and "Hide all" buttons to show all the templates or to hide them all, respectively.

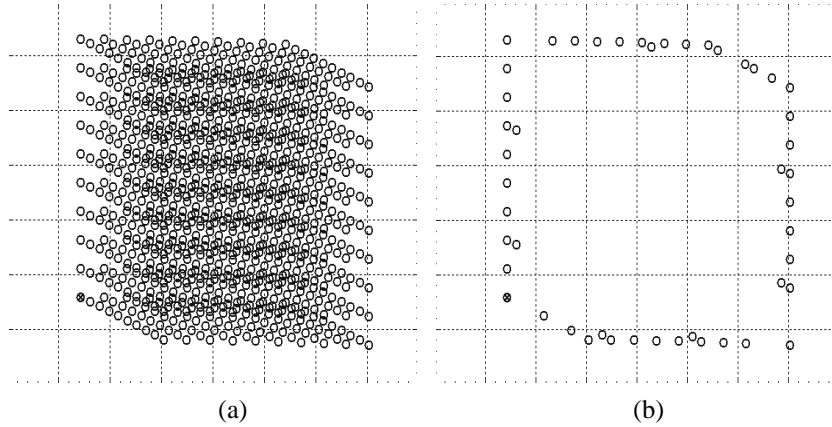


Fig. F.17 Template without contour adjustment [1000 points, (a)], and with contour adjustment [51 points, (b)].

- **Modify parametric uncertainty** The user can modify the parametric uncertainty by pressing the “Modify uncertainty” button (Fig. F.15). Then, an emerging window appears (see Fig. F.18), where the user can modify the parametric uncertainty in the same way as in the plant definition window. The user can see how the changes in the parametric uncertainty are applied in real time to the templates window.

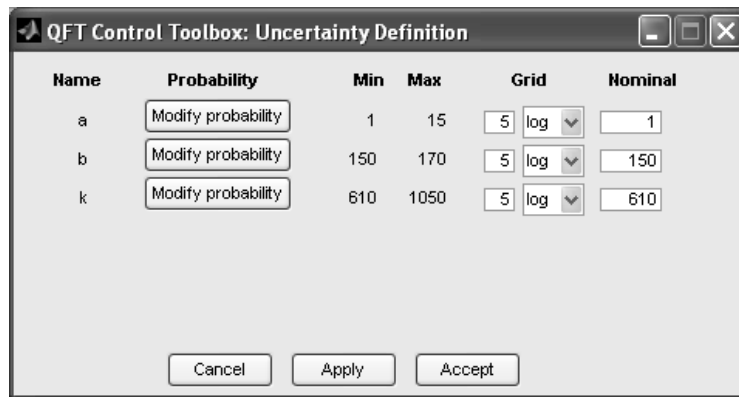


Fig. F.18 Parameter uncertainty window.

- **Export Nominal Plant:** The templates window has a menu item that allows the user to export the nominal plant as a transfer function. The nominal plant is saved in the current directory in a *.mat file. Afterwards, the user can load that file to the Matlab's Workspace (load *.mat) to use it.

F-2.4 SPECIFICATIONS WINDOW

This window allows the user to introduce robust stability and performance control specifications (see Fig. F.19). In the panel “Choose the specification type” to select seven groups of specifications. The first six options are the classical specifications [corresponding to Eqs. (F.6) to (F.10)], and the last one is a very general one [corresponding to Eq. (F.11)], able to generate the first five specifications and many other possibilities, including additional transfer functions like $M(s)$ in Fig. F.1 or others. See Table F.2 for more details.

- **Predefined Specifications:** [see Eqs. (F.6) to (F.10)]. The user can add specifications on any single-loop closed-loop relation (see Fig. F.1).

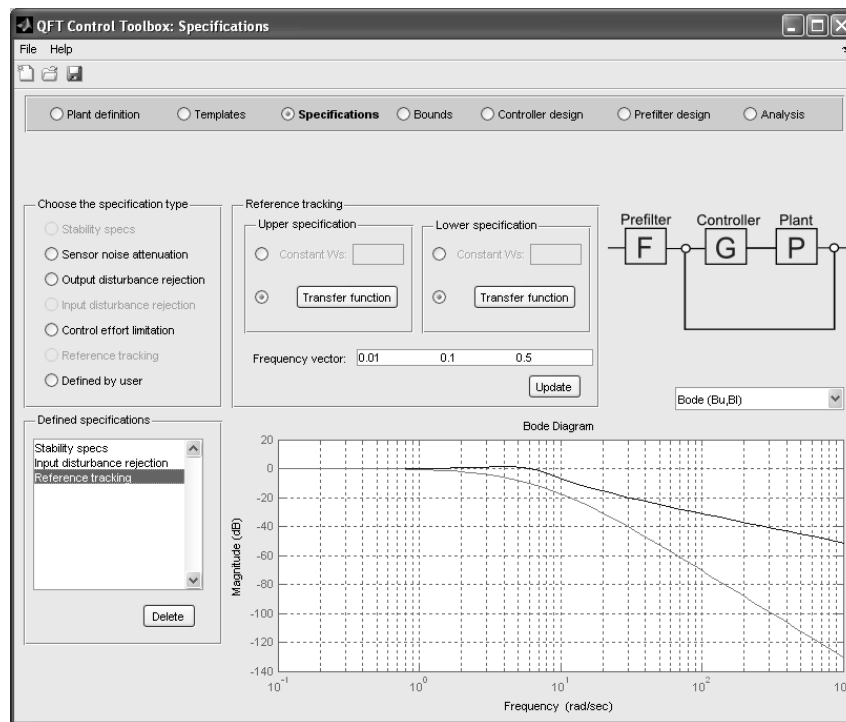


Fig. F.19 Performance specifications window.

- **User-defined specifications:** [see Eq. (F.11)]. All the pre-defined specifications, except the reference tracking, can be expressed using the user-defined specifications. Moreover, the user can use these specifications in the

design of cascaded-loop and multiple loops, and in systems that involve single-loop design at each design step.

Table F.2 Control system specifications.

$ T_1(j\omega) = \left \frac{Y(j\omega)}{R(j\omega)F(j\omega)} \right = \left \frac{U(j\omega)}{D_1(j\omega)} \right = \left \frac{Y(j\omega)}{N(j\omega)} \right = \left \frac{Y(j\omega)}{R(j\omega)F(j\omega)} \right =$ $= \left \frac{P(j\omega)G(j\omega)}{1 + P(j\omega)G(j\omega)} \right \leq \delta_1(\omega), \omega \in \{\omega_1\}$	(F.6)
$ T_2(j\omega) = \left \frac{Y(j\omega)}{D_2(j\omega)} \right = \left \frac{1}{1 + P(j\omega)G(j\omega)} \right \leq \delta_2(\omega), \omega \in \{\omega_2\}$	(F.7)
$ T_3(j\omega) = \left \frac{Y(j\omega)}{D_1(j\omega)} \right = \left \frac{P(j\omega)}{1 + P(j\omega)G(j\omega)} \right \leq \delta_3(\omega), \omega \in \{\omega_3\}$	(F.8)
$ T_4(j\omega) = \left \frac{U(j\omega)}{D_2(j\omega)} \right = \left \frac{U(j\omega)}{N(j\omega)} \right = \left \frac{U(j\omega)}{R(j\omega)F(j\omega)} \right =$ $= \left \frac{G(j\omega)}{1 + P(j\omega)G(j\omega)} \right \leq \delta_4(\omega), \omega \in \{\omega_4\}$	(F.9)
$\delta_{5\inf}(\omega) \leq T_5(j\omega) = \left \frac{Y(j\omega)}{R(j\omega)} \right = \left F(j\omega) \frac{P(j\omega)G(j\omega)}{1 + P(j\omega)G(j\omega)} \right \leq \delta_{5\sup}(\omega), \omega \in \{\omega_5\}$	(F.10)
$\left \frac{A(j\omega) + B(j\omega)G(j\omega)}{C(j\omega) + D(j\omega)G(j\omega)} \right \leq \delta_6(\omega), \omega \in \{\omega_6\}$	(F.11)

Looking at Fig. F.1 and Table F.2, it is seen that:

- Eq. (F.6) $[T_1(j\omega)]$ defines three types of specifications: (1) robust stability, (2) robust control effort limitation from the input disturbance, and (3) robust sensor noise attenuation.
- Eq. (F.7) $[T_2(j\omega)]$ defines two type of specifications: (1) robust rejection of disturbances at the output of the plant, and (2) the sensitivity.
- Eq. (F.8) $[T_3(j\omega)]$ defines one type of specification: robust rejection of disturbances at the input of the plant.
- Eq. (F.9) $[T_4(j\omega)]$ defines three types of specifications: robust control effort limitation from (1) the output disturbance, (2) the sensor noise, and (3) the filtered reference signal.

- Eq. (F.10) $[T_5(j\omega)]$ defines one type of specification: robust reference tracking.
- Eq. (F.11) $[T_6(j\omega)]$ defines any specification from type $T_1(j\omega)$ to $T_4(j\omega)$, and many other options in general, where $A(j\omega)$, $B(j\omega)$, $C(j\omega)$, and $D(j\omega)$ can be defined by the user, from several options like 0, 1, $P(j\omega)$, or any other plant introduced in the Plant definition window (for example $M(j\omega)$ to define the specification $|M(j\omega)/[1+P(j\omega)G(j\omega)]|$).

The value of $\delta_i(\omega)$ denotes the magnitude of the objective (the specification) at every frequency of interest. Each specification can be defined for a different set of frequencies ω , always a sub-set of the original set of frequencies of interest.

- **Defining a Specification** The user has to enter the value of the performance specification $\delta_i(\omega)$ [$\delta_{\text{inf}}(\omega)$ and $\delta_{\text{sup}}(\omega)$ in the reference tracking case]. The method to enter the value of $\delta_i(\omega)$ depends on the type of specification being defined, so that:

- **Robust stability specs** [see Table F.2, Eq. (F.6) and Fig. F.20]. $\delta_1(\omega)$ is a constant (Ws). The user can enter either directly the value of $\delta_1(\omega) = \mu$ –in magnitude–, which represents the M -circle in the Nichols diagram (in this case the gain and phase margin are calculated automatically), or the gain GM or phase PM margins (in that case μ is calculated automatically). The following equations are used in the calculations:

Gain Margin: $GM \geq 1 + (1/\mu)$ (magnitude)

Phase margin: $PM \geq 180^\circ - \theta$ (deg)

where: μ is the M circle specification in magnitude, $M_{dB} = 20\log_{10}(\mu)$,

and $\theta = 2\cos^{-1}(0.5/\mu) \in [0, 180^\circ]$

- **Robust performance specs.** [see Table F.2, Eqs. (F.6) to (F.9) and Figs. F.21 and F.22]. $\delta_i(\omega)$, $i = 1, 2, 3, 4$ can be defined as a constant (see Fig. F.21), a vector of constants of the same length as the specification frequency vector, or as transfer function (zero/pole/gain or num/den, see Fig. F.22). In this case, if $\delta_i(\omega)$ is a transfer function, its Bode diagram is also displayed in the Specification window (similar to Fig. F.19).

Fig. F.20 Defining stability specifications.

Fig. F.21 Defining $\delta_i(\omega)$, $i = 1,2,3,4$, as a constant.

Fig. F.22 Defining $\delta_i(\omega)$, $i = 1,2,3,4$, as a transfer function.

- **Robust reference tracking specs.** [see Table F.2, Eq. (F.10) and Figs. F.23 and F.24]. In this case the user has to define $\delta_{\text{inf}}(\omega) = \delta_{\text{lower}}(\omega)$ and $\delta_{\text{sup}}(\omega) = \delta_{\text{upper}}(\omega)$ (see Fig. F.23). Again, both values can be defined as constants, as a vector of constants or as transfer functions. Besides, if this type of specification is selected, there are more plots available: the step response in the time domain of $\delta_{\text{lower}}(j\omega)$ and $\delta_{\text{upper}}(j\omega)$, the Bode diagram of $\delta_{\text{upper}}(j\omega)$ - $\delta_{\text{lower}}(j\omega)$, see Fig. F.23, and the Bode diagram of $\delta_{\text{lower}}(j\omega)$ and $\delta_{\text{upper}}(j\omega)$, see Fig. F.19.

Fig. F.23 Defining $\delta_{\text{upper}}(\omega)$ and $\delta_{\text{lower}}(\omega)$ specifications.

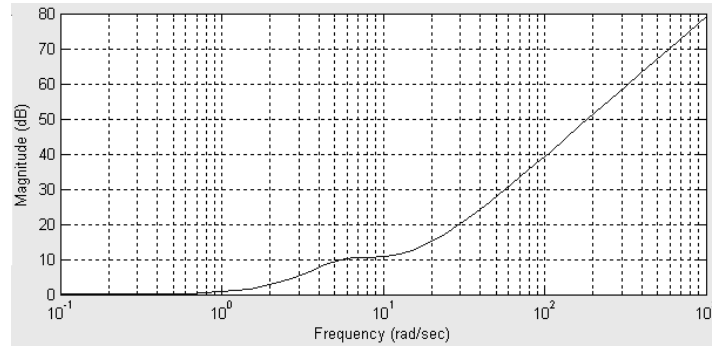


Fig. F.24 Bode diagram of $\delta_{\text{upper}}(\omega)$ - $\delta_{\text{lower}}(\omega)$.

- **Defined by user specs.** [see Table F.2, Eq. (F.11)]. In this case the user not only has to enter $\delta_i(\omega)$, but also $A(j\omega)$, $B(j\omega)$, $C(j\omega)$, and $D(j\omega)$, for a specification like: $\|[A(j\omega)+B(j\omega)G(j\omega)]/[C(j\omega)+D(j\omega)G(j\omega)]\| < \delta_i(\omega)$. They can be constants (0 or 1), the system plant $P(j\omega)$ or other auxiliary plants defined in the Plant Definition window [$G(j\omega)$ is the controller].

- **Frequency Vector** The user has to define a frequency vector for each specification. This vector must be a subset of the frequency vector of the system plant.

Specification Addition The user has to select in the “Choose specification type” panel, see Fig. F.19 or F.25, the type of specification to be defined. Then the user can enter the parameters of the specification. When the data are entered the user presses the “Commit” button to add the specification.

Fig. F.25 Defining “Defined by user” specs., Eq. (F.11), for example having $M(j\omega)$ from Plant definition window, and doing $|M(j\omega)/[1+P(j\omega)G(j\omega)]|$.

- **Specification Edition** To edit a specification, the user has to select it in the “Defined specifications” panel. Then all the data of the specification will be displayed in the center panel of the window. Once the specification has been edited, the user has to press the “Update” button to apply the changes.
- **Specification Removal** To remove a specification the user has to select it in the “Defined specifications” panel and press the “Delete” button.

F-2.5 BOUNDS WINDOW

Given the plant templates and the control specifications, QFT converts closed-loop magnitude specifications $[T_1(j\omega) \text{ to } T_6(j\omega)]$, into magnitude and phase constraints for a nominal open-loop function $L_0(j\omega)$. These constraints are called QFT bounds (see Fig. F.26). After the design of the controller (next section), the nominal open-loop function $L_0(j\omega)$ must remain above the solid-line bounds and below the dashed-line bounds at each specific frequency, to fulfill the desired specifications.

The Bounds window (see Fig. F.26) shows the QFT bounds of each specification defined in the Specification window, as well as the Union of all the bounds and the Intersection (worst case scenario) of all the bounds.

By clicking “Show all” or “Hide all”, the toolbox plots all the bounds or hides all the bounds. By double clicking on its corresponding frequency, the bound of that frequency is shown or hidden.

The user can also define the minimum, maximum and step value of the phase vector by using the “Edit phase vector” submenu in the “Edit menu”.

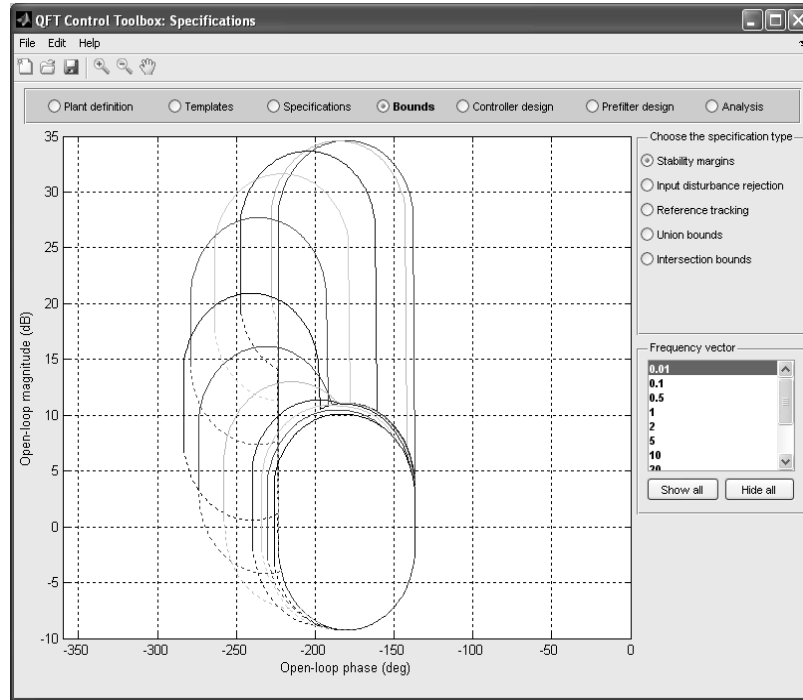


Fig. F.26 Bounds window.

F-2.6 CONTROLLER DESIGN WINDOW

Once the user has introduced the information about the plant and the control specifications, and once the templates and bounds have been calculated, the next step involves the design (loop shaping) of the controller $G(s)$, so that the nominal open-loop transfer function $L_0(s) = P_0(s) G(s)$ meets the bounds (see Fig. F.27). Generally speaking, the loop shaping, or $G(s)$ design, involves changing the gain and adding poles and zeros, either real or complex, until the nominal loop $L_0(s)$ lies near its bounds, more specifically above the solid-line bounds and below the dashed-line bounds at each frequency of interest.

- **Controller Management**

- **Controller addition** There are two ways of add a new controller:
 - To create the new controller from scratch, the user has to enter the name of the controller and press the “Add controller” button. If there is not any other controller defined with that name, the new controller appears in the list of added controllers.
 - If the user wants to add a new controller based of the dynamics of an existing controller, the user has to select it from the “Added controllers” list and press the “Copy” button. The user has to enter the name of the new controller in the emerging window.

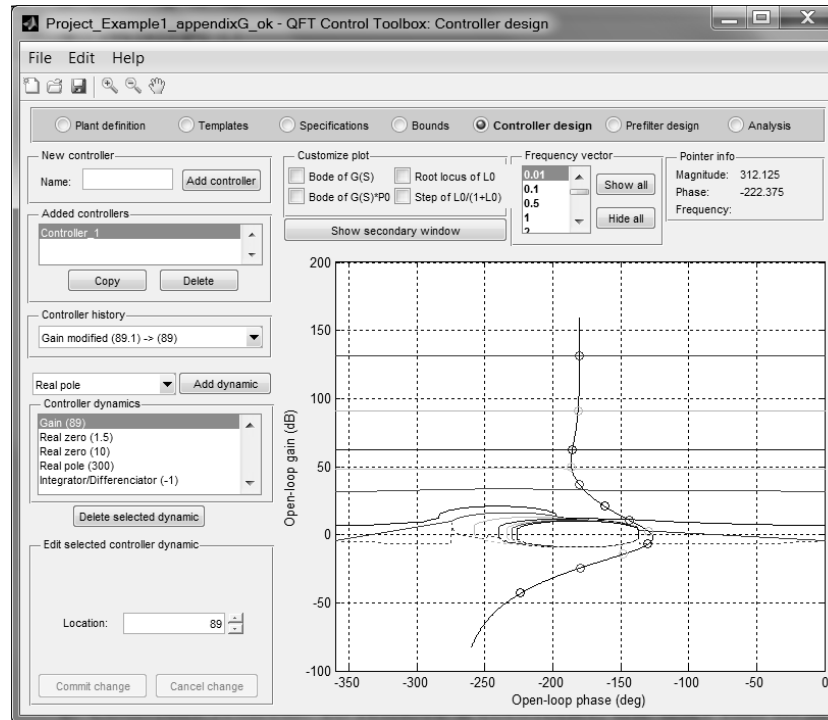


Fig. F.27 Controller Design window.

- **Controller removal** To remove a controller, the user has to select it from the list of “Added controllers” and press the “Delete button”.

- **Controller dynamics** When defining a controller, the user can work with the dynamics listed in the Table F.3, according to Eq. (F.12).

$$L_0(s) = P_0(s) \prod_{i=0}^n [G_i(s)] \quad (\text{F.12})$$

To add a new dynamic element $G_i(s)$ in the controller $G(s) = \prod G_i(s)$, the user has to press the “Add dynamic” button. The dynamic is added with its predefined values. In the lower left corner of the window there is a panel which shows information about the selected dynamic. The information displayed depends on the type of the element, as shown in Figs. F.28 to F.31.

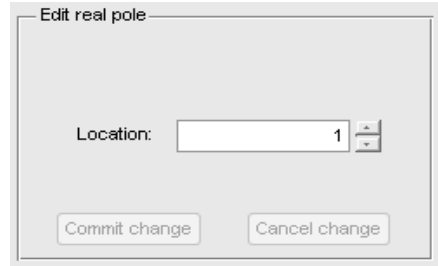


Fig. F.28 Window for Gain, zero, pole and integrator/differentiator.

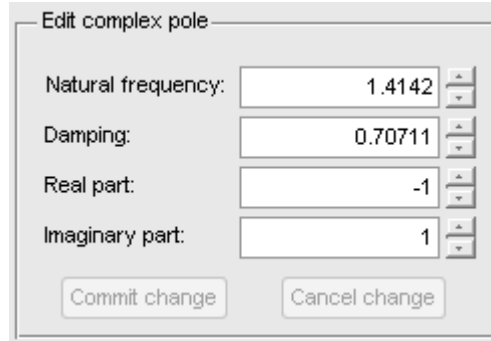
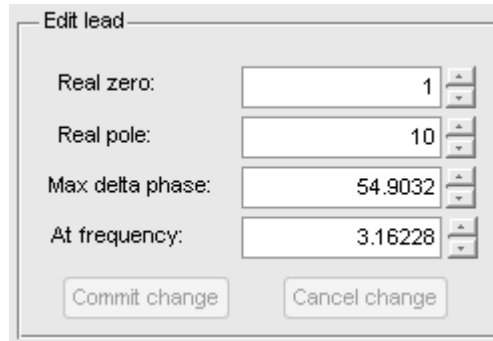


Fig. F.29 Window for Complex zero and complex pole.

Table F.3 Controller elements, $G_i(s)$.

Gain	k	2° order / 2° order	$\frac{a_1 s^2 + a_2 s + 1}{b_1 s^2 + b_2 s + 1}$
Real pole	$\frac{1}{(s/p) + 1}$	Integrator	$\frac{1}{s^n}$
Real zero	$(s/z) + 1$	Differentiator	s^n
Complex pole	$\frac{1}{(s/\omega_n)^2 + (2\zeta/\omega_n)s + 1}$ ($\zeta < 1$)	Lead/lag network	$\frac{(s/z) + 1}{(s/p) + 1}$
Complex zero	$(s/\omega_n)^2 + (2\zeta/\omega_n)s + 1$ ($\zeta < 1$)	Notch filter	$\frac{(s/\omega_n)^2 + (2\zeta_1/\omega_n)s + 1}{(s/\omega_n)^2 + (2\zeta_2/\omega_n)s + 1}$

**Fig. F.30** Window for Lead/Lag element.

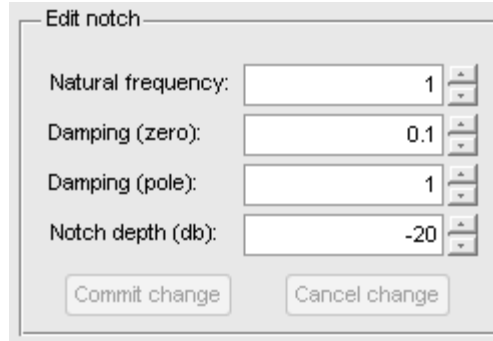


Fig. F.31 Window for Notch filter.

○ **Dynamic edition:**

▪ **With textual tools** All the dynamics can be edited by entering their new values in the textboxes that appear in the “Edit selected controller dynamic” panel (see Fig. F.27), as shown in Figs. F.28 to F.31.

▪ **With graphical tools** The dynamic elements $G_i(s)$ of the controller can be edited by interacting with the graphs with the mouse. This is a very powerful tool that helps the user in the controller design. To use it, first the element has to be selected in the “Controller dynamics” panel. For example:

- the *gain* can be modified by (1) dragging vertically the $L_0(j\omega)$ line with the mouse in the Nichols chart; or (2) by dragging vertically the Bode magnitude plot of $G(s)$ in the secondary window; or (3) by dragging vertically the Bode magnitude plot of $G(s) P_0(s)$ in the secondary window.

- a *zero* or a *pole* can be modified by (1) dragging the $L_0(j\omega)$ line to the right or to the left with the mouse in the Nichols chart; or (2) by dragging to the right or to the left the selected zero or pole in the Bode magnitude plot of $G(s)$ in the secondary window; or (3) by dragging to the right or to the left the selected zero or pole in the Bode magnitude plot of $G(s) P_0(s)$ in the secondary window.

○ **Dynamic removal**

The user can remove an element of the controller by selecting it in the “Controller dynamics” panel and pressing the “Delete selected dynamic” button.

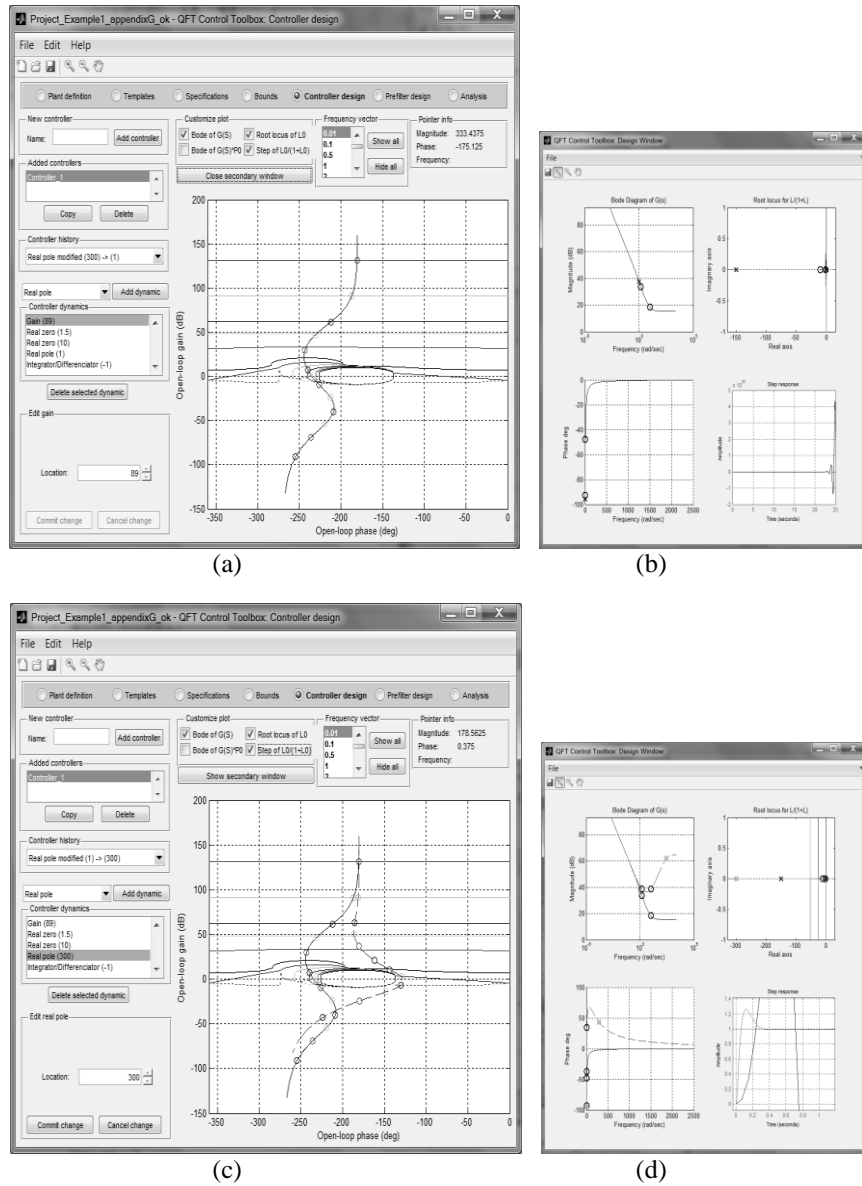


Fig. F.32 An example of the graphic dynamic edition: (a) Controller design window –Nichols chart–, (b) Secondary window –Bode diagram, Root locus and Step time response–, (c) Controller design window after moving a pole from $p = 1$ to $p = 300 \{1/[(s/p) + 1]\}$, (d) Secondary window after moving that pole.

Mario Garcia-Sanz, Augusto Mauch, Christian Philippe, et al. (2011). *The QFT Control Toolbox (QFTCT) for MATLAB*. Case Western Reserve University. <http://csc.case.edu>

Mario Garcia-Sanz, Constantine H. Houppis, (2012), *Wind Energy Systems: Control Engineering Design*, CRC Press, Taylor & Francis. ISBN: 978-1-4398-2179-4. (Appendix F: *QFT Control Toolbox User's Guide*).

When the user edits some dynamics of the controller, the updated $L_0(j\omega)$ is shown in the Nichols plot as a red dashed line (Figure F.32c), meaning that the changes are provisional. To commit the changes the user has to press the “Commit button”. Then the red dashed line is drawn as a solid black line. To discard the changes, the user has to press the “Cancel button”.

Figs. F.32(a) to F.32d illustrate an example of the graphic dynamic edition. The Figs F.32(a) and F.32(b) show the Controller design window and the secondary window. To meet the bounds, the user changes the position of the selected real pole, from $p = 1$ to $p = 300$. The user can do it by entering the new value in the textbox, or by dragging horizontally the green ‘x’ mark in the bode plot. The Figs F.32(c) and F.32(d) show how all the plots have been updated accordingly.

- **Change History** Each time the user adds, edits or removes a dynamic element, the change is added to the Controller history panel (see Fig. F.33). The user can undo and redo changes by selecting different entries in the Controller history panel, by pressing Ctrl+x and Ctrl+y respectively, and by using the Edit menu as well.
- **Secondary Window** The user can open a secondary window with the following plots (see Figs. F.32 and F.34): Bode diagram of $G(s)$, Bode diagram of $L_0(s)$, root locus of $L_0(s)$ and unit step response of $L_0(s)/[1+L_0(s)]$. The secondary window can be resized, and the changes made to the selected controller are applied in real time to all the diagrams.

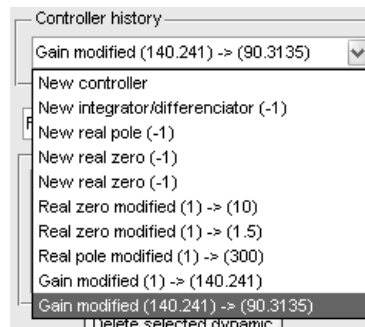


Fig. F.33 Controller history panel.

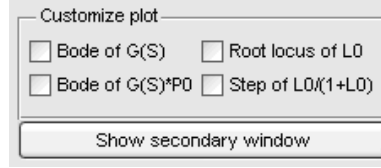


Fig. F.34 Panel to open de secondary window.

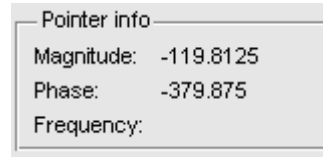


Fig. F.35 Pointer info panel.

- **Pointer Information** In the upper right corner of the window there is a panel that shows information about the pointer (see Fig. F.35). If the pointer is over the Nichols plot, the panel shows information about where the pointer is (magnitude and phase). If the pointer is placed over the $L_0(j\omega)$ line, the panel also includes the frequency ω associated with that point.

F-2.7 PRE-FILTER DESIGN WINDOW

If the control problem requires reference tracking specifications, then the Pre-filter design window is active after the design of the $G(s)$ controller.

Fig. F.36 shows the Pre-filter design window, with the following plots: the upper and lower reference tracking specifications $[\delta_{upper}(\omega), \delta_{lower}(\omega)$, dashed lines], [see Eq (F.10) and Fig. F.23], and the maximum and minimum cases of $L_0(s)F(s)/[1+L_0(s)]$ over the plant uncertainty (lines between δ_{upper} and δ_{lower}).

The design of the pre-filter involves in obtaining the worst upper and lower closed-loop response cases of $L_0(s)F(s)/[1+L_0(s)]$ over the plant uncertainty, which should be between the upper and lower reference tracking functions to meet the specifications. The Pre-filter Design window is very similar to the Controller Design window. The way in which the pre-filters are added, edited and removed is the same.

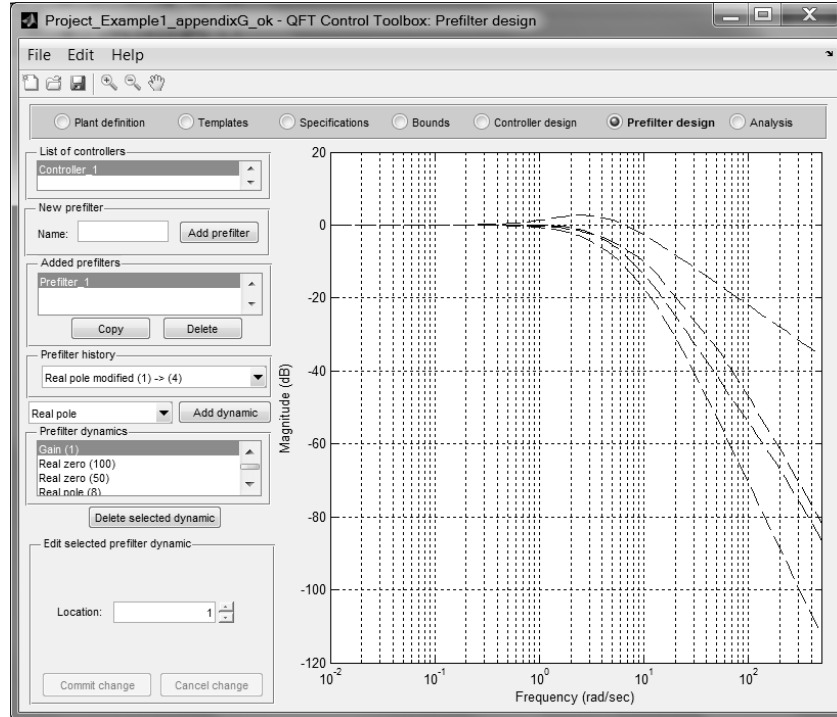


Fig. F.36 Pre-filter Design window.

The Pre-filter Design window has an additional “List of controllers” panel which allows the user to select among the controllers designed in the previous window.

F-2.8 ANALYSIS WINDOW

Once the user finishes the controller (and pre-filter) design, the Analysis window is active. The analysis is performed in both, the frequency domain and the time domain. The window analyses the controller $G(s)$ and pre-filter $F(s)$ in the worst case scenario over the plant uncertainty.

- The window allows the user to perform two types of analysis:
 - **Frequency Domain Analysis Panel** This panel permits an analysis of the closed-loop response of the control system with respect to a specification defined in the specification window.

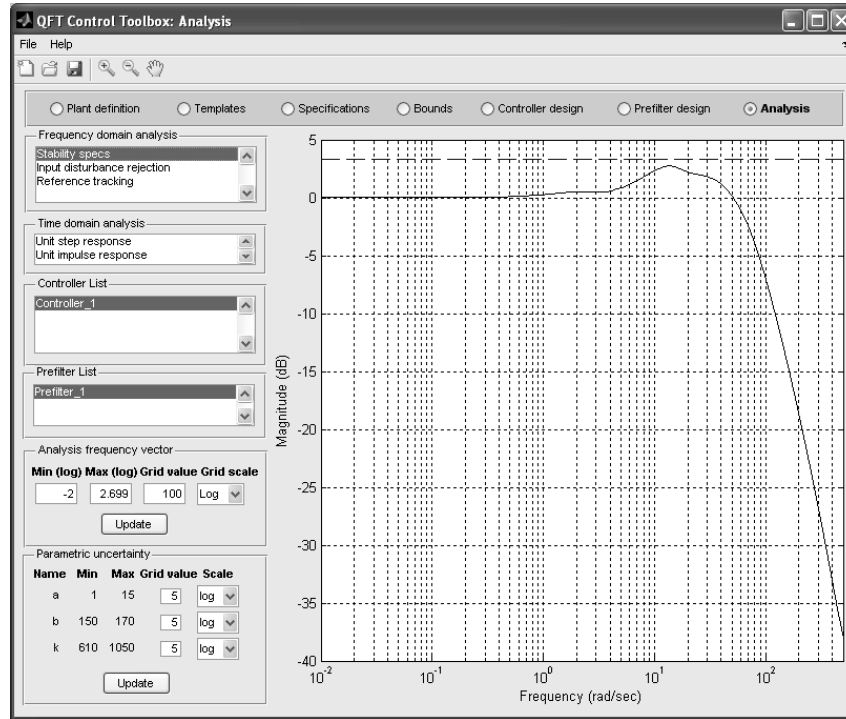


Fig. F.37 Analysis window.

The dashed line is the desired specification $\delta_i(\omega)$ and the solid line the worst case of the control system over the plant uncertainty at each frequency (see Fig. F.37 for the analysis of the stability specification in the frequency domain). Note that the solid line is not a transfer function, but the worst case among all the transfer functions at every frequency over the plant uncertainty.

- **Time Domain Analysis** This panel analyses the time response of the control system, with many plants over the plant uncertainty (see Fig. F.38 for the analysis of the unit step response in the time domain). The window can apply:

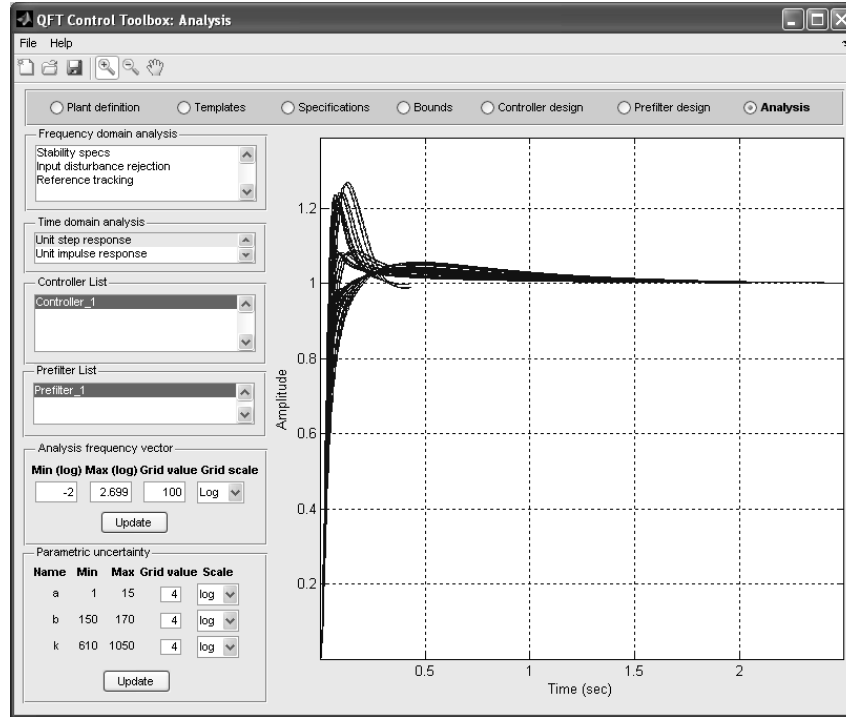


Fig. F.38 Time domain analysis.

- (1) a unit step at the reference signal, studying the performance of $L_0(s)F(s)/[1+L_0(s)]$ or $L_0(s)/[1+L_0(s)]$; and
- (2) a unit impulse at the output of the plant, studying the performance of $1/[1+L_0(s)]$.

The number of plants analyzed (number of lines plotted) depends on the values introduced in the “Parametric uncertainty” panel (see Fig. F.38).

- **Controller/Pre-filter Combinations** The user can select any combination of controllers (from the Controller List panel) and pre-filters (from the Pre-filter List panel) previously defined (if any).
- **Frequency Vector Panel for the analysis** The Analysis frequency vector panel allows the user to enter the frequency vector to be used in the frequency domain analysis. If there are not enough points in the frequency vector, the resulting analysis may not be accurate enough (see Figs. F.39 and F.40).

- **Parametric Uncertainty Panel** As it was explained, this panel allows the user to modify the grid of the parametric uncertainty variables. Again, if too few points are selected, the analysis may not be accurate enough. On the other hand, if too many points are selected, the analysis may be slow. This panel allows the user to analyze the system in both, (1) the points previously defined in the plant definition window (with the parametric uncertainty panel), and (2) in new points of uncertainty, defined now in the analysis window with the parametric uncertainty panel.

If the responses for all the *selected plants* satisfy the desired control performance specifications at both, frequency and time domains, then the design is completed. If the design fails at any frequency or time, you may decide to re-design the control (and pre-filter), or to re-define the plant model, uncertainty, or specifications definition.

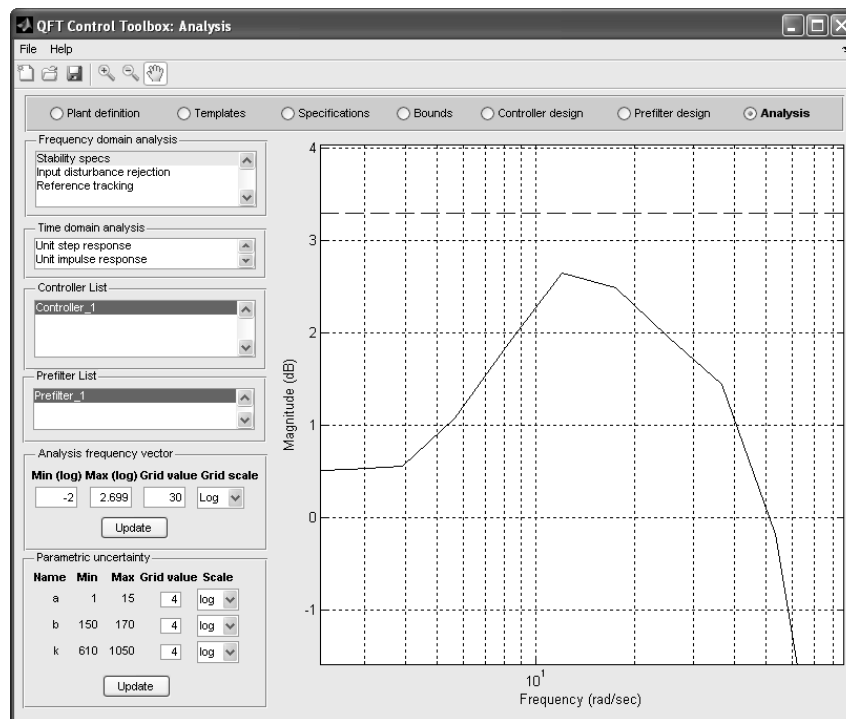


Fig. F.39 Stability analysis with not enough points in the frequency vector.

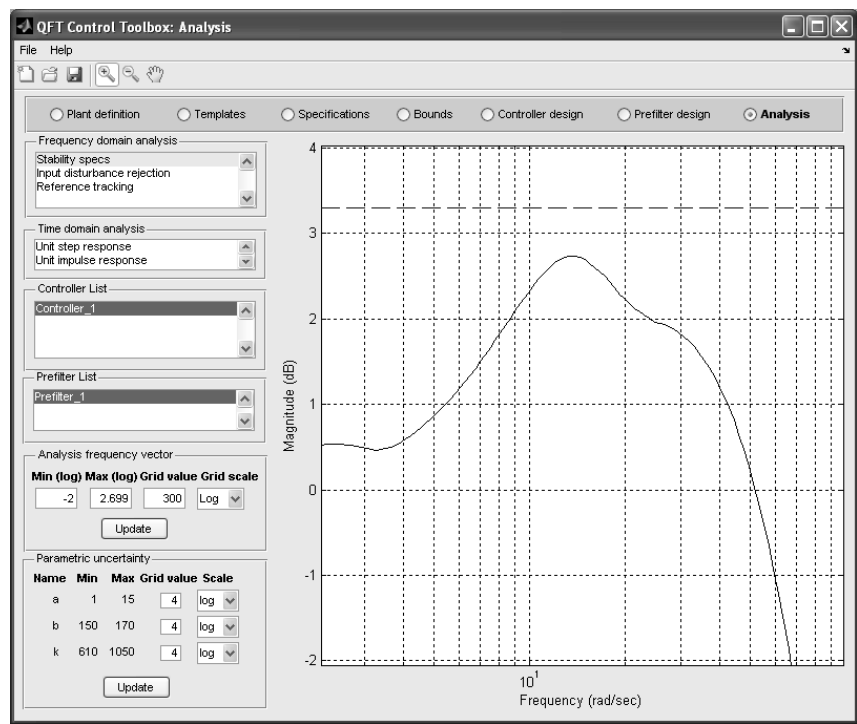


Fig. F.40 Stability analysis with a more populated grid.