International Carpathian Control Conference ICCC' 2006
Rožnov pod Radhoštěm, CZECH REPUBLIC
May 29-31, 2006

Fractional-order PID controller design on Internet: www.PIDlab.com

Martin ČECH[1] and Miloš SCHLEGEL[2]
[1] Department of Cybernetics,
University of West Bohemia in Pilsen
Pilsen, Czech Republic, mcech@kky.zcu.cz
[2] schlegel@kky.zcu.cz

Abstract: The fractional-order PID controller (FPID) arises from the classical one after considering integrator and derivator of arbitrary real order. This paper describes the simple Internet tool for FPID design - the Java applet free accessible on www.PIDlab.com. The applet implements procedures from the author's previous work where the robustness regions method for frequency domain FPID design was presented. The implemented procedure allows to define several frequency domain closed loop requirements simultaneously (gain and phase margins, limitations of sensitivity functions, proper bandwidth) Moreover, the applet allows FPID designing from experimental data (e.g. one sample of the frequency response, moments of the impulse response) using model set approach. We believe that this Internet tool can be helpful not only for researcher and students but also for industrial sphere.

Key words: fractional-order systems, PID controller, robustness regions method, Java applet

1 Introduction

Over the years, the PID controller becomes one of the most popular industrial controllers thanks to its simplicity and the ability to tune its few parameters automatically. The boom of the fractional calculus in the last decades showed new possibilities for process identification [Charef 1992] and controller synthesis [Podlubny 1999]. In this area, the fractional-order PID controller (FPID) arises from the classical one after considering integrator and derivator of arbitrary real order. Moreover, these two parameters still have a clear physical interpretation. The generalized fractional order operators are also linear, hence the same methods and rules for frequency domain linear control system design can be applied.

This paper describes the simple Internet tool for FPID design - the Java applet free accessible on www.PIDlab.com. The applet implements procedures from the author's previous work [Schlegel 2006], where the robustness regions method for frequency domain FPID design was presented.

The paper is organized as follows: Section 2 briefly introduces fractional-order systems and the FPID controller. Basic facts about robustness regions method are reminded in Section 3. The short user description of the Java applet is given in Section 4. Illustrative example is shown in Section 5. Concluding remarks and ideas for further work can be found in Section 6.

2 Fractional order systems and FPID controller

From the identification point of view, fractional-order transfer functions enable to express arbitrary fractional slopes of Bode plots of real processes (temperature, chemical). Using simple fractional operators in the controller synthesis, one has wider Nyquist plot shaping possibilities. The simple fractional-order integrator is described by the transfer function

$$\frac{1}{s^a}, \quad a \in R. \tag{1}$$

Adding integrator (1) to the control loop can help to reach the ideal Nyquist plot shape proposed by Bode. Similarly, the fractional order derivator can be defined.

Substituting integer-order operators by fractional-order ones, the FPID controller with transfer function

$$C(s) = K\left(1 + \frac{1}{T_i s^a} + T_d s^b\right), \quad a, b \in R \tag{2}$$

grows up. For a practical digital realization, the filter has to be added to the derivative part. Then the 1DOF FPID is described by

$$C(s) = K\left(1 + \frac{1}{T_i s^a} + \frac{T_d s^b}{\frac{T_d}{N}s + 1}\right). \tag{3}$$

It was shown [Schlegel 2004], that the filter parameter N has a significant influence to the other parameters and must be taken into account. Further, let us denote $K_i = K / T_i$.

3 Robustness regions method

It is well known, that required loop performance (e. g. gain and phase margins) could be reached by shaping of the Nyquist plot

$$L(\mathrm{j}w) = C(\mathrm{j}w)P(\mathrm{j}w). \tag{4}$$

For example, the minimum gain margin 2 (Gm>2) and minimum phase margin 60˚ ($Pm \geq$ 60˚) are equivalent with requirements that points X_1 = -1/2 and $X_2 = -1/2(1+\mathrm{j}\sqrt{3})$ lie on the left side of $L(\mathrm{j}\omega)$ or just on this curve. General choice of shaping points X can ensure limitations of sensitivity function, specified disturbance rejection at a given frequency range and the proper bandwidth of the closed loop.

Solving the equation

$$C(\mathrm{j}w)P(\mathrm{j}w) = X, \quad w \in (0, \infty) \tag{5}$$

for fixed controller parameters α, β, N and $f = T_i / T_d$ one obtains a curve in parameter plane K-K_i (K_i=K/T_i) parameterized by ω. This curve together with k, k_i axis splits the parametric plane into several regions. Usually just one region contains suitable points corresponding to the required relative location of the Nyquist plot and the point X. However, four controller parameters have to be determined ***a priori*** to draw robustness regions. Fortunately, parameter N has the clear physical interpretation. When $N \to \infty$, we obtain an ideal FPID controller, while the derivative part is switched off if $N \to 0$. The value of N is usually chosen in the interval $\langle 1, 10 \rangle$ according to the signal/noise ratio. Moreover, the ratio $f = T_d / T_i$ is used to be equal to ¼ [Åström 1995]. There are no general rules for the choice of the integrator order α and derivator order β. This choice is dependent on the frequency domain closed loop requirements. However, reasonable values are $a, b \in \langle 0, 2 \rangle$. To make a FPID design procedure routinely applicable, we decided to create an interactive Java applet.

4 Applet user description

One of the features of the Java applet *Fractional PID laboratory* free accessible at www.pidlab.com is the FPID controller design. In the applet, the robustness regions method for the general FPID controller with filtered derivative part is implemented.

Firstly, the user has to define the process model in the *Process* tab using one of the four forms (SOPTD, num/den, zero/pole, Bode). After that, one can move to the *Controller* tab to define the design specifications – shaping points. Before it, it is necessary to switch the controller form to '*FPID*' and to determine four controller parameters α, β, N and $f = T_i / T_d$. The design specifications can be simply added by mouse click in the *Nyquist plot shaping* window. After clicking, the shaping point can be edited manually after choosing in the list or moved by mouse dragging. The M-circles limitations are also shown in the figure if the corresponding checkbox is selected.

When a new shaping point is defined a corresponding region appears in the *Robustness regions* window. If more than one shaping points are defined, we can activate/deactivate them by the checkbox on the right side of the list.

The orders α, β are determined in the *'Orders plane'* . They can be changed by mouse click or dragging the cross. The regions are updated continuously according to their dependences on integrator and derivator orders.

To complete the design procedure, the remaining controller parameters K and K_i have to be chosen by mouse click in the ***Robustness Regions*** window. If one wants to fulfill more design specifications simultaneously, the controller parameters must be located in the intersection of all robustness regions created by corresponding design specifications. It is very advantageous that all frequency responses are recomputed while the parameter cross is dragged by mouse. If the user wants to fulfill the proper bandwidth or to dump the low-frequency disturbances he can define the shaping point directly in *CSF* or *SF* plot, respectively.

Described concept of mutually joined plots leads to the maximum interactivity and minimizes the overall duration of the design procedure.

5 Illustrative example

Lets us use the applet for a FPID controller design for a real temperature process described by the transfer function

$$P(s) = \frac{125.7}{(8.1s+1)^4(420.2s+1)}. \qquad (6)$$

The process model should be defined in *Process* tab as shown in Fig 2.

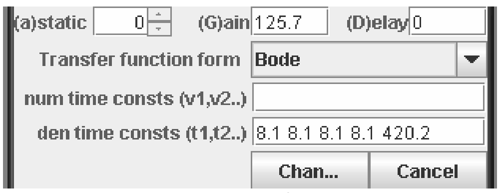

Fig. 2. Model definition

Let us assume that the sensitivity function upper limit is $M_s \le 1.4$ and the complementary sensitivity function upper limit is $M_T \le 1.6$. Corresponding *M*-circles are drawn if the checkbox is selected. After that we can change the parameters N and f (default values N=10, f=0.25 are recommended). The next step is the shaping points definition. To reach the ideal Bode shape of the Nyquist plot two shaping points close to the *M*-circles boundary as shown in Fig. 3.

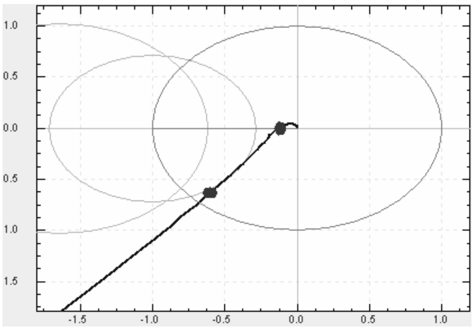

Fig.3. Shaping points and final Nyquist plot

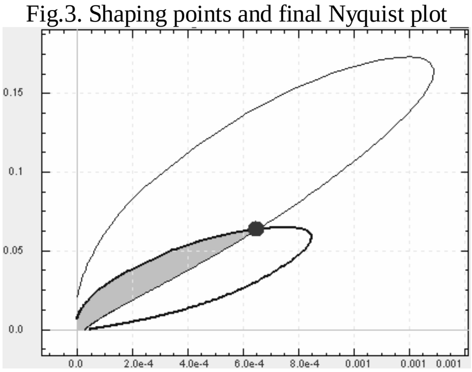

Fig.4. Robustness regions and the choice of the parameters *K, K*

To fulfil both design specifications, the controller parameters must be chosen in the intersection of regions (area highlighted in Fig. 4.). The fastest controller is the point with maximal K_i coordinate. The overall result of the example can be reviewed in Fig.1. Here one can also persuade that the sensitivity function does not exceed the required upper limit.

6 Conclusions

The simple Internet tool for frequency domain FPID controller design – the Java applet free accessible at www.PIDlab.com - was briefly described in this paper. It was shown that the two additional parameters – orders of integrator and derivator bring wider Nyquist plot shaping possibilities. It was demonstrated how the

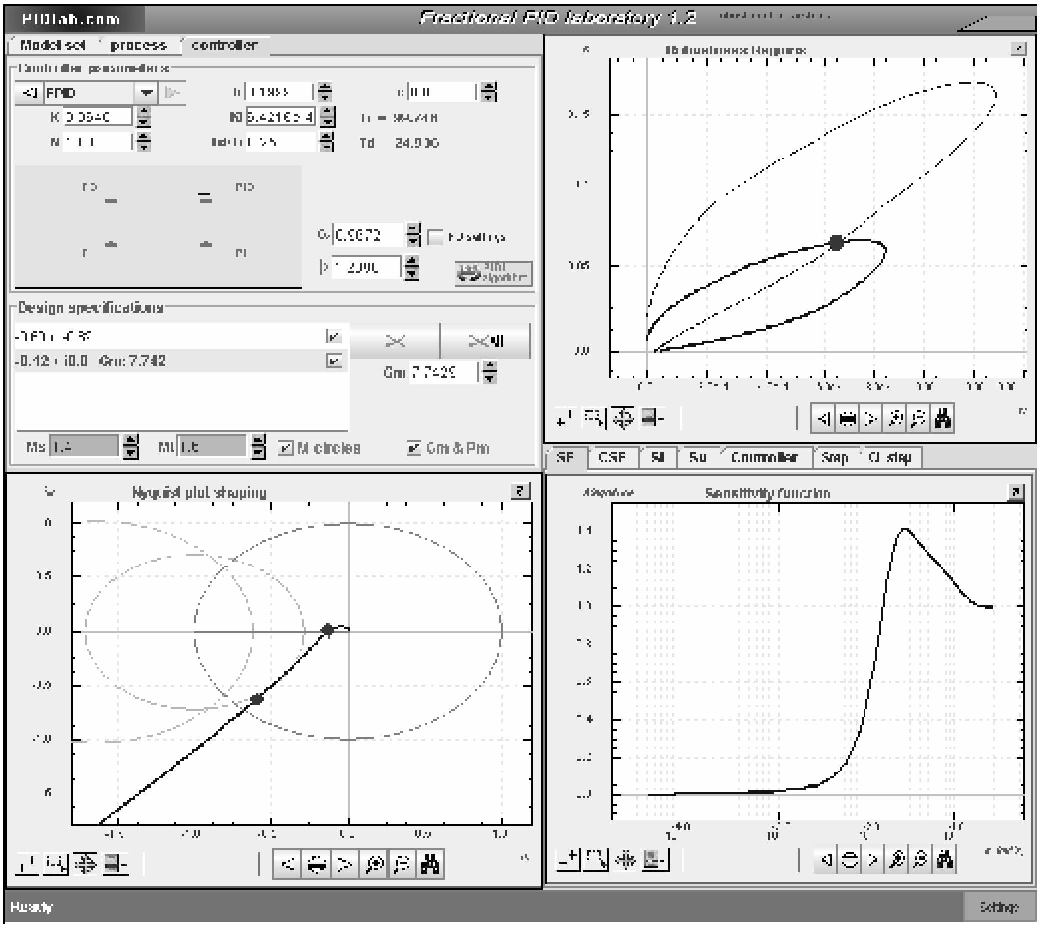

Fig.1. Applet view

generalized robustness regions method can be used to determine FPID parameters by the interactive Java applet. In the future, it will be necessary to create a reliable and robust digital FPID realization to use the developed procedures in real-time applications.

References

CHAREF, A.C, SUN, H.H., 1992. Fractal system as represented by singularity function. In. *IEEE Transactions on Automatic Control* 37:1465–1470, 1992.

PODLUBNY, I. 1999. Fractional-order systems and $PI^{\lambda} D^{\mu}$ controllers. *IEEE Transactions on Automatic Control*, 44:208–214, 1999.

SCHLEGEL, M., ČECH, M. 2004. Internet PID controller design: www.pidlab.com . In. *Proceedings of IBCE 2004 conference*, Grenoble, France

SCHLEGEL, M., ČECH, M. 2006. Fractional-order PID controller overperforms the classical one. In. *Proceedings of Process Control 2006 conference*, Kouty n. Desnou, Czech Republic.

ÅSTRÖM, K. J; HÄGGLUND, T. 1995. *PID controller. Theory design and tuning*. NC: Instrument Society of America, Research Triangle Park, 1995.

http://www.PIDlab.com – virtual Internet PID controller laboratory