Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Understanding PID design through interactive tools

J.L. Guzmán * **T. Hägglund** ** **K.J. Åström** ** **S. Dormido** ***
M. Berenguel * **Y. Piguet** ****

* *University of Almería, Almería, Spain. {joguzman,beren}@ual.es*
** *Lund University, Lund, Sweden. {tore,kja}@control.lth.se*
*** *UNED, Madrid, Spain. sdormido@dia.uned.es*
**** *Calerga Sàrl, La Tour-de-Peilz, Switzerland.*
yves.piguet@calerga.com

Abstract: Hundreds of PID design methods are available in literature. Many of them are very similar and sometimes it is not straightforward to understand their purposes. This paper presents an interactive software tool to help in the study and understanding of several well-known PID tuning rules. Frequency- and time-domain responses are analyzed in order to show the robustness and performance properties of each method. Furthermore, a free tuning option is available to provide comparisons with user-defined rules or other existing tuning methods. A wide range of stable process models are included in the tool, where model-reduction techniques are applied for high-order processes to obtain simple models for design purposes.

Keywords: PID control, tuning rules, control design, control education, interactivity.

1. INTRODUCTION

PID control is by far the control strategy most widely used in industry (Åström and Hägglund, 2005). Besides its simplicity, complex control problems can be solved by using just a PID-based feedback control loop and/or by complementing the loop with simple additional filters (Hägglund, 2013). Because of its high impact in industry, PID tuning has received an important attention by researchers along the years, resulting in hundreds of available tuning methods (O'Dwyer, 2003). This variety of methods ranges from simple methods (mainly oriented to people with a limited knowledge in control) where the PID parameters can be obtained just by entering some parameters characterizing the process dynamics in a given set of formulae (Ziegler and Nichols, 1942; Cohen and Coon, 1953), to those based on standard analytical techniques and variants (Dahlin, 1968; Rivera et al., 1986; Skogestad, 2003; Grimholt and Skogestad, 2012), or some others based on optimization methods (Panagopoulos et al., 2002; Hägglund and Åström, 2004).

This huge availability of design methods makes the tuning task (by the choice of the appropriate method) sometimes complicated and confusing. This problem is specially remarkable from an educational point of view, being complex for students during the learning stage and for industrial practitioners during training periods. Traditionally, this issue has been solved by making multiple simulations and analyzing the different results. However, this solution is time consuming and lots of plots and information are required to analyze and study all possible situations: changes in model parameters, modifications of the control specifications, etc.

Thus, based on the above motivation and according to the authors' experience in control education and interactivity, the objective of this work is to present a new interactive tool as a support to control education for the PID design problem. The tool includes four tuning methods (Zielger-Nichols, λ tuning, AMIGO, and SIMC), which have been selected as the most representative ones from an industrial point of view (Åström and Hägglund, 2005). A free-tuning option is also available to provide manual tuning or to include some other tuning rules for comparison. The tool has been focused on the load disturbance rejection problem, since similar set-point tracking responses can be obtained among the different methods by using a reference filter or setpoint-weighting ideas. Performance and robustness measurements are provided, and the reaction curve method is used as model reduction technique in order to obtain low-order models for design purposes.

The new tool presented in this paper belongs to a set of interactive tools that has been developed in the last years under the Interactive Modules Learning (ILM) project. The idea of this project emerged during the writing of the book Åström and Hägglund (2005) and consists in developing interactive software tools which could be used for introductory control courses at universities and for engineers in industry. The modules are self-contained, they are suitable for self-study, for courses, and for demonstrations in lectures, and they do not require any additional software. Currently, with this new tool about PID design, most of the contents available in the book (Åström and

⋆ This work has been funded by the National Plan Project DPI2011-27818-C02-01 of the Spanish Ministry of Science and Innovation and FEDER funds, and by the Swedish Foundation for Strategic Research through the PICLU center.

Hägglund, 2005) have an associated interactive tool as support: Modelling and identification (Guzmán et al., 2008a), PID control (Guzmán et al., 2008b), dead-time control (Guzmán et al., 2008c), interaction (Guzmán et al., 2009), and feedforward (Guzmán et al., 2011); covering so a wide range of automatic control concepts.

The paper is organized as follows. In Section 2, a short summary of the PID design methods included in the tool is given. The interactive tool is presented in Section 3, and a few examples are described in Section 4. The paper ends with conclusions and references.

2. PID DESIGN METHODS

In this work, the classical feedback control problem described in Figure 1 is studied, where it is assumed that the feedback controller is a PI or PID controller with transfer function

$$C(s) = K\left(1 + \frac{1}{sT_i} + sT_d\right), \tag{1}$$

with K being the proportional gain, T_i the integral time, and T_d the derivative time, and where $T_d = 0$ for PI control (which is the case considered in this paper).

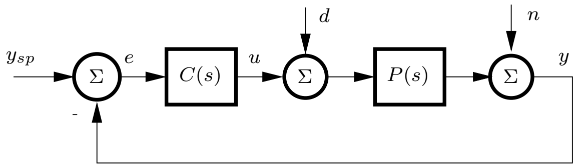

Fig. 1. Block diagram of the feedback control problem.

The process dynamics, $P(s)$, is approximated by a first-order process with time delay (FOPTD) model for design purposes,

$$\hat{P}(s) = \frac{k}{1+sT}e^{-sL} \tag{2}$$

where k is the static gain, $T > 0$ is the time constant, and L is the time delay. When $P(s)$ presents high-order dynamics, the reaction curve method is used to obtain a reduced model as (2) (Åström and Hägglund, 2005).

2.1 Ziegler-Nichols method

The Ziegler and Nichols method was originally formulated in 1942, (Ziegler and Nichols, 1942), and it is still widely used. It often forms the basis for tuning procedures used by controller manufacturers and the process industry. The method is based on obtaining simple features of the process dynamics and then to determine the controller parameters based on these features. Concretely, the Ziegler and Nichols method is based on process information in the form of the open-loop step response and is focused on lag dominant processes.

The step response is characterized by only two parameters, gain a and time L, as shown in Figure 2. Then, PID controller parameters are given directly as functions of a and L, as presented in Table 1.

The main features of the Ziegler-Nichols tuning rules are that they are simple and intuitive, they require little process knowledge, and they can be applied with modest

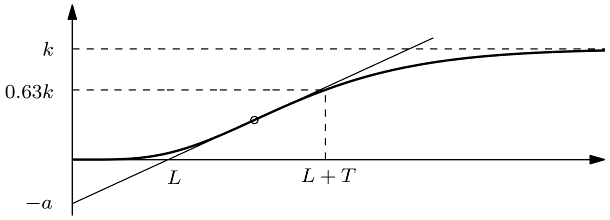

Fig. 2. Characterization of a step response in the Ziegler-Nichols step response method.

Table 1. Controller parameters for the Ziegler-Nichols step response method.

Controller	aK	T_i/L	T_d/L
P	1		
PI	0.9	3	
PID	1.2	2	1/2

effort. The method is based on the design criterion quarter amplitude damping, which often gives high performance in terms of a small integrated absolute error (IAE) value at load disturbances, but the main drawback of this method is that the closed-loop systems present poor robustness. These rules should be taken as an initial design stage that has to be followed by manual fine tuning.

2.2 Lambda method

Lambda tuning is a special case of pole placement that is commonly used in the process industry. The idea is to cancel the dominating process pole with the controller zero by setting integral time T_i equal to the time constant T. Approximating the time delay using the Taylor series expansion $e^{-sL} \approx (1 - sL)$, the loop transfer function becomes (Dahlin, 1968)

$$G_l(s) = C(s)P(s) \approx \frac{Kk(1-sL)}{sT} \tag{3}$$

Requiring that the closed-loop pole is $s = -1/T_{cl}$, where T_{cl} is the desired closed-loop time constant, we find

$$kK = \frac{T}{L + T_{cl}},$$

which gives the following simple tuning rule

$$\begin{aligned} K &= \frac{1}{k}\frac{T}{L+T_{cl}} \\ T_i &= T. \end{aligned} \tag{4}$$

The closed-loop response time T_{cl} is a design parameter. In the original work by Dahlin (1968) it was denoted as $T_{cl} = \lambda$, which explains the name lambda (λ) tuning.

The choice of T_{cl} is critical. A common rule of thumb is to choose $T_{cl} = 3T$ for a robust controller and $T_{cl} = T$ for aggressive tuning when the process parameters are well determined. Both choices lead to controllers with zero gain and zero integral time for pure time delay systems. For delay-dominated processes it is therefore sometimes recommended to choose T_i as the largest of the values T and $3L$.

Lambda tuning is a simple method that can give good results in certain circumstances provided that the design parameter is chosen properly. The basic method cancels a process pole which will lead to poor responses to load disturbances for lag-dominated processes.

2.3 SIMC method

Skogestad has developed a version of Internal Model Control (IMC) tuning method for PID control that avoids some of the drawbacks mentioned above for the Lambda method. In this method, it is required that the closed-loop system should have the transfer function (Skogestad, 2003)

$$G_{yysp} = \frac{1}{1+sT_{cl}}e^{-sL}$$

For an FOPTD system, the controller transfer function is

$$C(s) = \frac{1+sT}{k(1+sT_{cl}-e^{-sL})} \approx \frac{1+sT}{sk(T_{cl}+L)}$$

where the exponential function is approximated using a Taylor series expansion. In contrast with the recommendation for IMC, the closed-loop time constant is proportional to the time delay L. The choice $T_{cl} = L$ is recommended. The integral term is also modified for lag-dominated processes. The original tuning rule for PI control was

$$\begin{aligned} K &= \frac{1}{k}\frac{T}{T_{cl}+L} \\ T_i &= \min(T, 4(T_{cl}+L)) \end{aligned} \tag{5}$$

It was recently improved in (Grimholt and Skogestad, 2012) to increase the robustness properties resulting in the following rule

$$\begin{aligned} K &= \frac{1}{k}\frac{T+L/3}{T_{cl}+L} \\ T_i &= \min(T+L/3, 4(T_{cl}+L)) \end{aligned} \tag{6}$$

The tuning parameter T_{cl} should be chosen to get the desired trade-off between fast response (small IAE) on one side, and acceptable control signal activity and robustness (small maximum sensitivity value, M_s) on the other side. The recommended choice $T_{cl} = L$ gives robust (M_s about 1.6 to 1.7) and somewhat conservative settings when compared with most other tuning rules. If it is desirable to obtain faster control, one may consider reducing T_{cl} to about $L/2$. More commonly, one may want to have "smoother" control with $T_{cl} > L$ and a smaller controller gain K (Grimholt and Skogestad, 2012). In the tool presented in this paper, T_{cl} will be referenced as λ_S for this method.

2.4 AMIGO method

Åström and Hägglund developed simple tuning rules in the spirit of the work done by Ziegler and Nichols in the 1940s. The goal was to make rules that can be used both for manual tuning and in auto-tuners for a wide range of processes. The methods were developed by applying the techniques for robust loop shaping to a large test batch of representative processes. The controller parameters obtained were then correlated with simple features of process dynamics (Hägglund and Åström, 2004).

One interesting observations was that there are significant differences between processes with delay-dominated and lag-dominated dynamics. To capture this difference, process dynamics must be characterized by at least three parameters. Notice that Ziegler and Nichols used only two parameters. One possible choice is: process gain k, apparent time constant T, and apparent time delay L. These parameters can be obtained from a step response experiment.

The design goal was to optimize the performance at step changes in the load with robustness constraints determined by the combined sensitivity $M = 1.4$ ($M = M_s = M_t$, with M_t the complementary sensitivity). The following formulae were obtained for PI control

$$\begin{aligned} K &= \frac{0.15}{k} + \left(0.35 - \frac{LT}{(L+T)^2}\right)\frac{T}{kL} \\ T_i &= 0.35L + \frac{13LT^2}{T^2+12LT+7L^2}, \end{aligned} \tag{7}$$

The tuning rule gave good results for all processes in a large test batch ranging from process with integration to processes with pure time delay.

2.5 Rule-based empirical tuning

Since some tuning methods, like Ziegler-Nichols, give control designs with poor robustness properties, it is sometimes necessary to complement these methods by manual tuning to obtain reasonable closed-loop properties. Manual tuning is typically performed by experiments on the process in closed loop. A perturbation is introduced either as a set-point change or as a change in the control variable. The closed-loop response is observed, and the controller parameters are adjusted. The adjustments are based on simple rules which give guidelines for changing the parameters. The tuning is a compromise between performance, e.g. expressed in terms of the IAE value at load disturbances, robustness, e.g. expressed in terms of M values, and control signal activity, e.g. expressed in terms of controller gain K. The following is a simple set of rules of thumb (Åström and Hägglund, 2005):

- Performance is improved by increasing controller gain or decreasing integral time.
- Robustness is improved by decreasing controller gain or increasing integral time.
- Control signal activity is reduced by decreasing the controller gain.

3. INTERACTIVE TOOL

This section briefly describes the main features of the developed tool. The tool has been implemented in Sysquake, a Matlab-like language with fast execution and excellent facilities for interactive graphics (Piguet, 2013). Windows, Mac, and Linux operating systems are supported.

This interactive learning module has been developed following the same structure as the other available tools in the ILM project (Guzmán et al., 2008a,b,c, 2009). It is composed of several parts. Different menus are available to load examples and choose between different configurations of the tool. The graphics can be manipulated directly by dragging points, lines, and curves or by using text-edits and sliders. Notice also that for all the graphics available in the tool, the horizontal sliders and textedits embedded in the graphics allows modification of the plot scales.

As always considered when developing this kind of tools, one of the most important issues to have in mind is the organization of the main windows and menus of the tool to facilitate the understanding of the underlying concepts (Dormido, 2004). The main window of the tool is divided into several sections represented in Figure 3, which are described as follows:

- *Time-domain control graphics.* There exist two graphical elements on the right side of the screen, which represent the system output (Step Response Y/D) and the controller output (Step Response U/D). The graphics show the simulation results of the control algorithm selected for a unitary step change in load disturbance. There are five different checkboxes located between these two figures that allow to choose the desired tuning method to be simulated. These options are Free, to show the response of a free tuning design according to the tuning parameters located in the *Process and control parameters* section, and AMIGO, Z-N, Lambda, and Skogestad, to show the responses for AMIGO, Ziegler-Nichols, Lambda, and Skogestad methods, respectively.
- *Process and control parameters.* The different parameters available in the tool are shown on the left-hand side of the screen (see Figure 3). These elements allow to modify the process and controller parameters, respectively. So, it is possible to modify the parameters of the process transfer function, and for the feedback controller, the PI parameters can be changed interactively. The user can modify the proportional gain, K, and the integral time, T_i, for the free tuning option, and also change the closed-loop time constants for Lambda (λ) and Skogestad (λ_S) methods.
- *Performance and robustness parameters.* Some parameters to characterize performance and robustness for the different tuning methods are available. These indices are provided below the process and control parameters area for each selected control method. The resulting PI controller parameters for the different tuning methods are shown in this area, where the integral gain, $K_i = K/T_i$, is also included as an estimate of the load disturbance rejection performance. The load disturbance response is characterized by the integral absolute error IAE_d for disturbance rejection, and the M_s value is shown as a robustness measurement.
- *Modeling graphics.* These graphics are located below the performance and robustness parameters, in the middle of the left-hand side of the tool. There is a time-domain graphic showing the step response for the high-order model that represents the real process, and the reduced model obtained by using the reaction curve method for design purposes. In the same way, the transfer functions for both models are shown at the right part of this figure.
- *Nyquist graphics.* Two Nyquist plots are presented in the left-bottom part of the tool as a visual information about stability and robustness. The plot in the left-hand side shows a general view of the Nyquist curves, and the plot in the right-hand side presents an augmented view around the critical point -1.
- *Menu options.* Several menu options can be chosen from the menu bar to select and modify the model transfer functions and some simulation parameters.

Notice that different colors have been used in the tool to identify the information of the selected tuning methods: black for Free tuning, red for AMIGO, blue for Ziegler-Nichols, magenta for λ, and green for Skogestad.

Regarding the implementation issues of this tool, a novel fast step-based response implementation for continuous-time systems with time delay has been used. This method is based on the Laplace and Z transforms which exploits the system linearity, and permits to obtain very fast interactive simulations by running several control loops simultaneously (Piguet and Müllhaupt, 2008).

4. ILLUSTRATIVE EXAMPLES

This section briefly shows an example of how the interactive tool can be used to compare and understand the different tuning methods. Let's choose the following transfer function from the Model menu:

$$P(s) = \frac{k}{(s+1)^n} e^{-Ls}$$

where the default values for the process parameters are: $k = 2$, $n = 3$, and $L = 0.2$. Then the following reduced model is obtained by using the reaction curve method:

$$\hat{P}(s) = \frac{2}{1+2.5s} e^{-s}$$

which gives a good approximation as shown in Figure 3.

Once the process model is selected, the controller parameters for all the tuning methods are calculated automatically (if they are properly switched on from the checkboxes between the time-domain plots), and performance and robustness measurements, as well as the time-domain and frequency-domain responses for all of them, are depicted in the corresponding areas of the tool. For this example, $\lambda = T = 2.5$ and $\lambda_s = L = 1$ were used for the λ and Skogestad methods, respectively. Figure 3 shows the results for the different methods. As can be observed, the Ziegler-Nichols method gives the best results from a performance point of view, with an IAE value of 3.6. This fact is also corroborated by the large integral gain, K_i, which is a good measure of the load disturbance rejection performance. The load disturbance response to low frequencies can be approximated by s/K_i (Åström and Hägglund, 2005), and thus a higher integral gain will give better responses to load disturbances. This is the case for the Ziegler-Nichols method in this example, which provides the highest integral gain, $K_i = 0.33$.

However, as commented above, the main drawback of the Ziegler-Nichols method is its poor robustness. This fact is observed from its high M_s value ($M_s = 3.25$) and also from the Nyquist plots. Skogestad gives a reasonable tradeoff between performance and robustness. Notice that the time-domain responses (and thus the IAE values) are quite close to the Ziegler-Nichols one, but the robustness is much better with M_s values around 2. Lambda and AMIGO methods give similar results, with more conservative and robust results. The AMIGO method provides

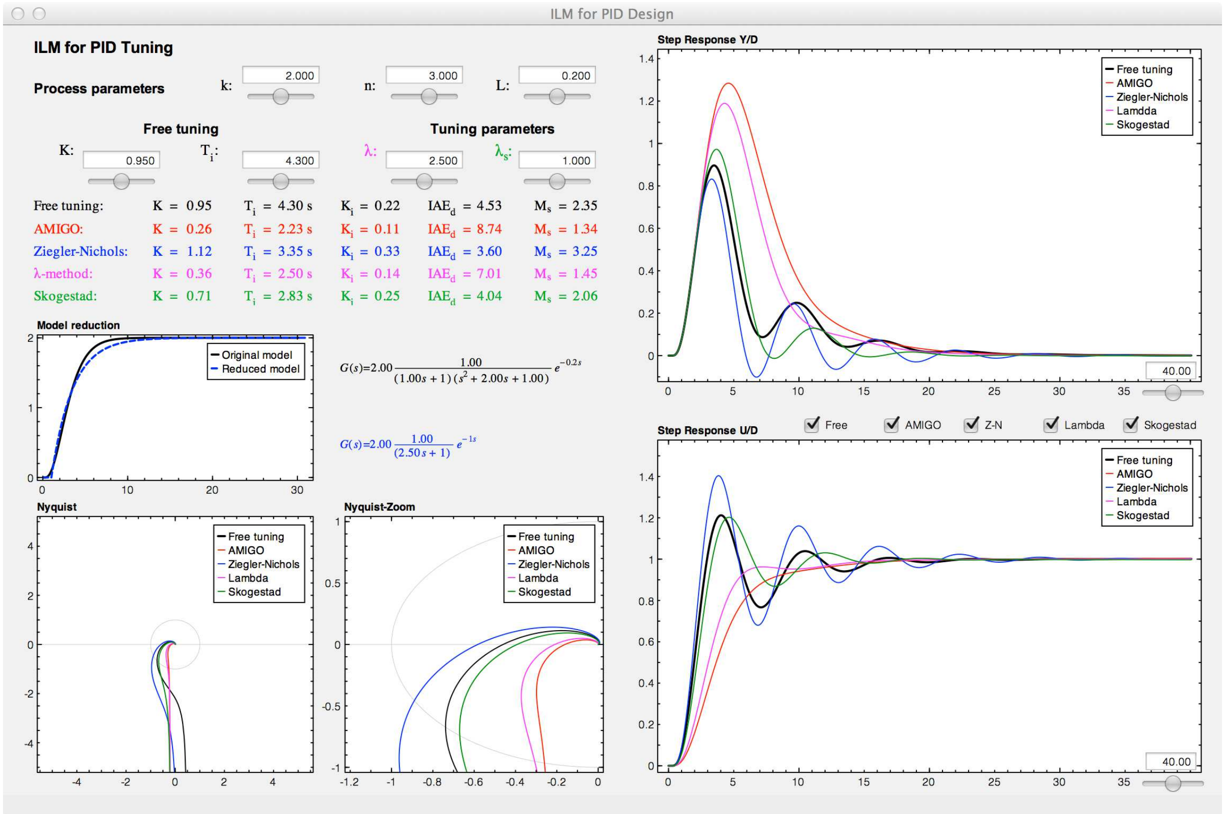

Fig. 3. Main window of the interactive tool. Example for a lag-dominant process.

the best robustness feature, but at expense of a more conservative time-domain response. Of course, this leads to a large IAE value, but an important issue for this method is that oscillations are avoided in the response, which is normally desired from an industrial point of view. The free tuning option can be used interactively to find out a result that e.g. is a compromise of the different methods. For instance, a tuning result with $K = 0.95$ and $T_i = 4.3$ gives an intermediate tuning between the more robust method (AMIGO) and the tuning rule with the best performance (Ziegler-Nichols), resulting in an IAE value of 4.53 and a M_s value of 2.35.

Notice that the previous example treated a lag-dominant process. The time delay of the process can be increased interactively to obtain a delay-dominant process, and one can observe how the different tuning methods behave for this new situation. Figure 4 shows the results for this case, where the time delay was increased from 0.2 to 5 (in this case $\lambda = 15$ and $\lambda_S = 5.8$ for the Lambda and Skogestad methods, respectively). As expected, the Ziegler-Nichols response is considerably deteriorated because of the long time delay (remember the condition $0.1 < L/T < 1$ for which this method works properly according to the quarter amplitude damping criteria). The Skogestad method seems to provide the best result based on a tradeoff between performance and robustness, although the AMIGO method gives similar values. The Lambda method gives more conservative results in this case. Notice that the AMIGO and Lambda methods provide non-oscillatory responses also in this case. In this example, the free tuning method can be used to improve the performance results, obtaining an IAE value of 20.28. Nevertheless, as is well known, the prize to improve the performance is a loss of robustness with a higher M_s value than in the other tuning methods.

5. CONCLUSION

A new interactive learning module for evaluating PID design methods has been described. The module belongs to the collection of interactive learning modules that is an attempt to make figures in the book (Åström and Hägglund, 2005) interactive. The tool allows evaluation of different PID tuning methods. The advantages and disadvantages of these methods for the regulation control problem can be studied interactively. Furthermore, a free tuning option is available. With this free tuning alternative, the user can modify the PID controllers parameters in order to beat the tuning methods and understand how these modifications must be done to find a tradeoff between performance and robustness. The module is implemented in Sysquake and is available for free on the web at `http://aer.ual.es/ilm/` in the PID design section.

REFERENCES

Åström, K.J. and Hägglund, T. (2005). *Advanced PID Control.* ISA - The Instrumentation, Systems, and Automation Society, Research Triangle Park, NC 27709.

Cohen, G.H. and Coon, G.A. (1953). Theoretical consideration of retarded control. *Transactions of ASAME*, 75, 827–834.

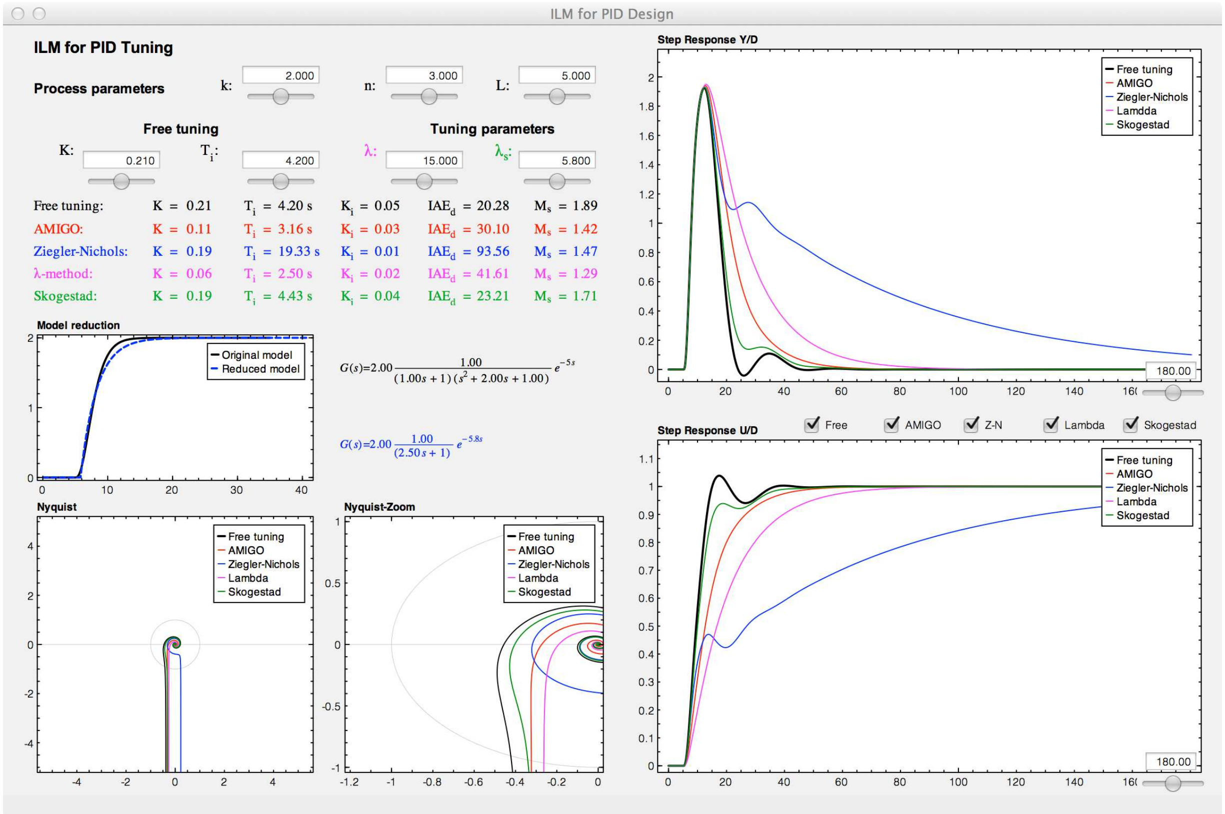

Fig. 4. Main window of the interactive tool. Example for a delay-dominant process.

Dahlin, E.B. (1968). Designing and tuning digital controllers. *Instruments and Control Systems*, 42, 77–83.

Dormido, S. (2004). Control learning: Present and future. *Annual Reviews in Control*, 28, 115–136.

Grimholt, C. and Skogestad, S. (2012). Optimal PI-Control and Verification of the SIMC Tuning Rule. *Proceedings of the IFAC Conference on Advances in PID Control PID'12*. Brescia (Italy).

Guzmán, J.L., Åström, K.J., Dormido, S., Hägglund, T., Berenguel, M., and Piguet, Y. (2008a). Interactive Learning Module: Basic Modelling and Identification Concepts. *Proceedings of the 17th IFAC World Congress*. Seoul (Korea).

Guzmán, J.L., Åström, K.J., Dormido, S., Hägglund, T., Berenguel, M., and Piguet, Y. (2008b). Interactive learning modules for PID control. *IEEE Control System Magazine*, 28(5), 118–134.

Guzmán, J.L., Åström, K.J., Dormido, S., Hägglund, T., Berenguel, M., and Piguet, Y. (2009). Interactive Learning Module for Control Interaction Understanding. *Proceedings of the European Control Conference*. Budapest (Hungary).

Guzmán, J.L., García, P., Hägglund, T., Dormido, S., Albertos, P., and Berenguel, M. (2008c). Interactive tool for analysis of time-delay systems with dead-time compensators. *Control Engineering Practice*, 16(7), 824–835.

Guzmán, J.L., Hägglund, T., Åström, K.J., Dormido, S., Berenguel, M., and Piguet, Y. (2011). Feedforward Control Concepts Through Interactive Tools. *Proceedings of the 18th IFAC World Congress*. Milano (Italy).

Hägglund, T. (2013). A unified discussion on signal filtering in PID control. *Journal of Process Control*, 21, 994–1006.

Hägglund, T. and Åström, K.J. (2004). Revisiting the Ziegler-Nichols step response method for PID control. *Journal of Process Control*, 14(6), 635–650.

O'Dwyer, A. (2003). *Handbook of PI and PID Controller Tuning Rules*. Imperial College Press, London.

Panagopoulos, H., Åström, K.J., and Hägglund, T. (2002). Design of PID controllers based on constrained optimization. *IEE Proceedings on Control Theory and Applications*, 149(1), 32–40.

Piguet, Y. (2013). *Sysquake 5 User Manual*. Calerga Sàrl, La Tour-de-Peilz, Switzerland.

Piguet, Y. and Müllhaupt, P. (2008). Fast Step-Response Evaluation of Linear Continuous-Time Systems with Time Delay in the Feedback Loop. *Proceedings of the 17th IFAC World Congress*. Seoul (Korea).

Rivera, D., Morari, M., and Skogestad, S. (1986). Internal Model Control 4. PID Controller Design. *Industrial & Engineering Chemistry Process Design and Development*, 25, 252–265.

Skogestad, S. (2003). Simple analytic rules for model reduction and PID controller tuning. *Journal of Process Control*, 13, 92–102.

Ziegler, J. and Nichols, N. (1942). Optimum settings for automatic controllers. *Transactions of ASAME*, 64, 759–768.