PID-Design-Delay: A MATLAB Toolbox for Stability Parameter-Space Characterization

Gao-Xia Fan[1], Yu-Jiao Chen[1], Xu-Guang Li[1], and Silviu-Iulian Niculescu[2]

***Abstract*—The proportional-integral-derivative (PID) controller design for time-delay systems is a classical problem. We consider a general case where the three controller gains (k_P, k_I, and k_D) and the delay (τ) are all treated as free parameters. Our objective is to characterize the stability set in the parameter space of (k_P, k_I, k_D, τ). The problem is very involved due to the theoretical difficulties. In this paper, we introduce a MATLAB-based toolbox, called PID-Design-Delay toolbox. It is developed based on some recently-reported theoretical results, by which the problem can be systematically solved. The user is only required to input the necessary information via the graphical user interface (GUI) and then the stability analysis may be automatically fulfilled by the toolbox.**

***Index Terms*—PID controller design, time-delay systems, MATLAB toolbox, GUI.**

I. Introduction

The proportional-integral-derivative (PID) controller design for time-delay systems is a rather common problem (see e.g., [16]), because on the one hand, PID controllers are used in more than 95% of industrial processes [1], and on the other hand, time-delay phenomena exist in almost all practical control systems (see e.g., [4] and [13]).

The problem considered in this paper is with a fairly general setup: The three controller gains (k_P, k_I, and k_D) and the delay (τ) are all treated as free parameters. Our objective is to find the stability set in the parameter space of (k_P, k_I, k_D, τ).

Most of the existing stability problems can be divided into the *τ-decomposition problem* [7] and the *D-decomposition problem* [14]. The former corresponds to the case where the delay is *free* and the other parameters are *fixed*. While, the latter corresponds to the counterpart case: The delay is *fixed* and some other parameters are *free*. We can see that the stability problem considered in this paper is a mixed τ-decomposition and D-decomposition problem.

The stability problem may be much more complicated than expected, due to the infinite-dimensional nature of time-delay systems (see e.g., [4] and [13]). A particular technical difficulty is related to the asymptotic behavior analysis for possible multiple and/or degenerate critical imaginary roots (CIRs) (see e.g., [8] and [10]).

Only a few toolboxes are available for stability analysis of time-delay systems (see e.g., TRACE-DDE [2], DDE-BIFTOOL [3], and QPmR [17], which are developed based on some numerical algorithms). Theoretical results for time-delay systems can be verified and better understood, with the aid of these toolboxes. The distribution of characteristic roots for a time-delay system can be nicely estimated by these toolboxes. However, to the best of the authors' knowledge, no toolbox has been reported so far for solving the PID stabilization problem of time-delay systems under consideration in this paper.

Recently, we developed a toolbox called *PID-Design-Delay*, based on the parameter-space approach proposed in [9][1]. By this toolbox, the PID controllers can be designed and the corresponding stability range of τ can be explicitly computed. It is worth mentioning that the *PID-Design-Delay* toolbox covers the general case: No constraint on the controlled plant is imposed, and arbitrarily complex asymptotic behavior of the CIRs is allowed.

The *PID-Design-Delay* toolbox is MATLAB based and simple to implement. The user is only required to input the necessary information via graphical user interface (GUI). This paper is dedicated to introduce how to solve the PID controller design problem by using this toolbox.

The remainder of this paper is organized as follows. Some preliminaries and prerequisites are given in Section II. In Section III, the basic operation of the toolbox is introduced. The functions of menu bars are explained in Section IV. In Section V, a case study is presented. Finally, Section VI concludes this paper.

II. Preliminaries and Prerequisites

Consider a closed-loop system consisted of a controlled plant and a PID controller as shown in Fig. 1.

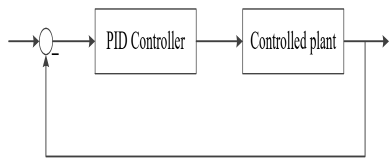


Fig. 1. Block diagram of PID control system

The controlled plant is described by the transfer function

$$H_0(\lambda) = \frac{H_{\mathrm{N}}(\lambda)}{H_{\mathrm{D}}(\lambda)}, \tag{1}$$

[1]G.-X. Fan, Y.-J. Chen, and X.-G. Li are with School of Information Science and Engineering, Northeastern University, Shenyang, Liaoning 110819, China fangaoxia0113@163.com, chenyujiao116@163.com, masdanlee@163.com

[2]S.-I. Niculescu is with University Paris-Saclays, CNRS, CentraleSupélec, Laboratoire des Signaux et Systèmes (L2S), Inria Team "DISCO", Gif-sur-Yvette, 91192, France Silviu.Niculescu@l2s.centralesupelec.fr

[1]More precisely, the theoretical development is based on the frequency-sweeping mathematical framework (see [8] and [10]). By using the *PID-Design-Delay* toolbox, the user may skip the theoretical part.

978-1-7281-9809-5/20/$31.00 ©2020 IEEE

where λ is the Laplace variable, $H_{\mathrm{D}}(\lambda)$ and $H_{\mathrm{N}}(\lambda)$ are coprime polynomials of λ with real coefficients. Without loss of generality, $H_{\mathrm{D}}(\lambda)$ and $H_{\mathrm{N}}(\lambda)$ can be described as

$$H_{\mathrm{D}}(\lambda) = a_0 + a_1\lambda + \cdots + a_{\mathrm{n}}\lambda^{\mathrm{n}}, a_{\mathrm{n}} \neq 0,$$
$$H_{\mathrm{N}}(\lambda) = b_0 + b_1\lambda + \cdots + b_{\mathrm{m}}\lambda^{\mathrm{m}}, b_{\mathrm{m}} \neq 0,$$

with $\mathrm{n} \geq \mathrm{m}$.

The control loop is usually subject to a delay τ, and thereby we can express the actual transfer function as $H_0(\lambda)e^{-\tau\lambda}$.

The PID controller is described by the transfer function

$$k_P + \frac{k_I}{\lambda} + k_D\lambda, \tag{2}$$

where k_P is the proportional gain, k_I is the integral gain, and k_D is the derivative gain.

The characteristic function for the closed-loop system is

$$f(\lambda, \tau) = H_{\mathrm{D}}(\lambda)\lambda + H_{\mathrm{N}}(\lambda)(k_I + k_P\lambda + k_D\lambda^2)e^{-\tau\lambda}. \tag{3}$$

The closed-loop system is asymptotically stable if and only if all the characteristic roots are located in the open left half-plane. In the sequel, we recall two useful notions.

If a system is asymptotically stable with

$$\tau \in [0, \bar{\tau}), \tag{4}$$

$\bar{\tau}$ is called the *delay margin* [4].

The notion of delay margin is widely used in stability analysis for time-delay system (see e.g., [5], [12], [11], and [15]). However, various examples are given in [9] to show that a plant may be stabilized with more than one interval including or excluding 0. In such cases, the notion is not valid. We now introduce a more general notion.

If a system is asymptotically stable with

$$\tau \in [\underline{\tau}_1, \bar{\tau}_1) \cup \cdots \cup (\underline{\tau}_s, \bar{\tau}_s), 0 = \underline{\tau}_1 < \bar{\tau}_1 < \cdots < \underline{\tau}_s < \bar{\tau}_s, \tag{5}$$

or

$$\tau \in (\underline{\tau}_1, \bar{\tau}_1) \cup \cdots \cup (\underline{\tau}_s, \bar{\tau}_s), 0 \leq \underline{\tau}_1 < \bar{\tau}_1 < \cdots < \underline{\tau}_s < \bar{\tau}_s, \tag{6}$$

$\bar{\tau}_s$ is called the *generalized delay margin* [9].

The *PID-Design-Delay* toolbox can be used to characterize the stability set in the parameter space of (k_P, k_I, k_D, τ), and the delay margin and generalized delay margin can be computed. The user is only required to provide some necessary information via the graphical user interface (GUI). Then, the *PID-Design-Delay* toolbox may automatically fulfill the stability analysis. The stability results will be displayed in the MATLAB command window as well as in 'Stability Set' figure.

If $k_D = 0$, the PID controller is reduced to a proportional-integral (PI) controller; if $k_I = 0$, the PID controller is reduced to a proportional-derivative (PD) controller.

The *PID-Design-Delay* toolbox includes three modes: PI, PD, and PID.

III. Basic operation of *PID-Design-Delay* toolbox

A. Starting PID-Design-Delay toolbox

The whole package is freely available at http://faculty.neu.edu.cn/ise/lixuguang/PIDCDTDS.html, where the manual and demo videos are also provided. The interested reader may also directly contact us via e-mail at masdanlee@163.com.

Unzip the downloaded package. Enter 'PIDCDTDS' in the command window of MATLAB (provided that the folder of the toolbox is added to MATLAB path or current folder). Alternatively, open 'PIDCDTDS.m' file in MATLAB, and press 'Run' button. Then, click 'Change Folder' button or 'Add to Path' button, if the user is prompted that the toolbox folder is not in the MATLAB path.

When the toolbox is started, the dialog box as shown in Fig. 2 will appear for the user to choose the type of controller (PI, PD, or PID). According to the user's choice, the dialog box for the PI mode (Fig. 3(a)), the PD mode (Fig. 3(b)), or the PID mode (Fig. 3(c)) will pop up.

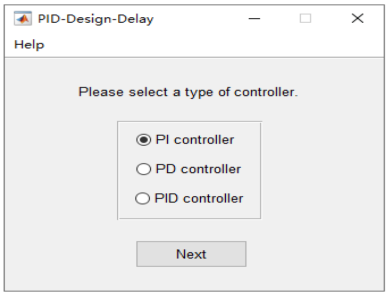


Fig. 2. Dialog box for selecting controller type.

B. Stability analysis in PI mode

If 'PI controller' is selected, the dialog box as shown in Fig. 3(a) will appear.

The PI controller is described by the transfer function

$$k_P + \frac{k_I}{\lambda}, \tag{7}$$

and the characteristic function of the closed-loop system is

$$f(\lambda, \tau) = H_{\mathrm{D}}(\lambda)\lambda + H_{\mathrm{N}}(\lambda)(k_I + k_P\lambda)e^{-\tau\lambda}. \tag{8}$$

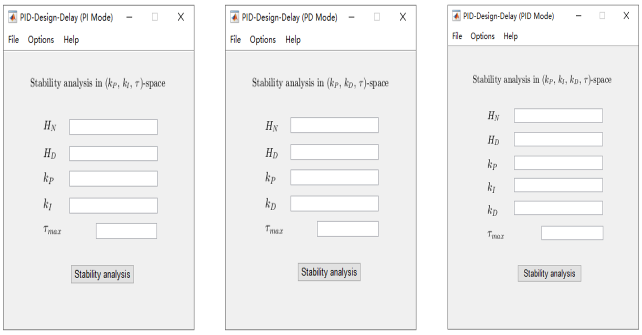
The user needs to provide some necessary information in the dialog box as shown in Fig. 3(a) as follows.

H_{N} edit box: Enter the coefficients of the numerator of controlled plant, i.e., the coefficients of $H_{\mathrm{N}}(\lambda)$. The input is in the format "$[b_{\mathrm{m}} \ldots b_0]$".

H_{D} edit box: Enter the coefficients of the denominator of controlled plant, i.e., the coefficients of $H_{\mathrm{D}}(\lambda)$. The input is in the format "$[a_{\mathrm{n}} \ldots a_0]$".

k_P edit box: Enter a range of k_P and the associated step length. The input is in the format "start point:step length:end point".

k_I edit box: Enter a range of k_I and the associated step length. The input is in the format "start point:step length:end point".



(a) PI mode (b) PD mode (c) PID mode

Fig. 3. Dialog boxes for inputting necessary information

τ_{max} edit box: Enter a positive value for τ_{max}. The delay τ is treated as a free parameter with $\tau \in [0, \tau_{max}]$ by the toolbox.

Remark 1: A tooltip string (Fig. 4) will pop up when the mouse pointer is on 'Stability analysis' button. The message is useful for the first-time users and will be explained by an example (Example 2).

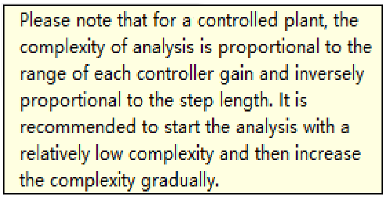


Fig. 4. Tooltip string associated with 'Stability analysis' button

After all the above information is appropriately provided, click 'Stability analysis' button. The *PID-Design-Delay* toolbox will take some time to fulfill the analysis. Finally, the stability results will be displayed in the MATLAB command window as well as in 'Stability Set' figure.

There are two cases that the PI controller definitely cannot stabilize.

In the case $\deg(H_{\mathrm{D}}(\lambda)) < \deg(H_{\mathrm{N}}(\lambda))$, the closed-loop system is of the advanced type if the PI controller is used. Thus, the closed-loop system definitely cannot be asymptotically stable. Accordingly, some indications will be given in the command window.

In the case $\deg(H_{\mathrm{D}}(\lambda)) = \deg(H_{\mathrm{N}}(\lambda))$, the closed-loop system is of the neutral type if the PI controller is used. A necessary condition for the system stability is

$$|k_P| < |\frac{a_{\mathrm{n}}}{b_{\mathrm{m}}}|. \tag{9}$$

If the above necessary condition is violated, some indications will be displayed in the command window accordingly.

Example 1: Apply the PI controller to the following controlled plant

$$\frac{\lambda - 2}{\lambda - 0.5}.$$

Suppose that the stability in the domain $(k_P, k_I, \tau) \in [-0.99, -0.01] \times [-1.5001, -0.0001] \times [0, 100]$ is of interest, and the step lengths for k_P and k_I are both to be set as 0.01.

The user should fill H_{N} edit box with "[1 -2]", H_{D} edit box with "[1 -0.5]", k_P edit box with "-0.99:0.01:-0.01", k_I edit box with "-1.5001:0.01:-0.0001", and τ_{max} edit box with "100".

After clicking 'Stability analysis' button, the toolbox will automatically perform the analysis. Then, the stability results will be displayed in the command window as well as in 'Stability set' figure. The application of *PID-Design-Delay* toolbox to Example 1 is shown in Fig. 5. □

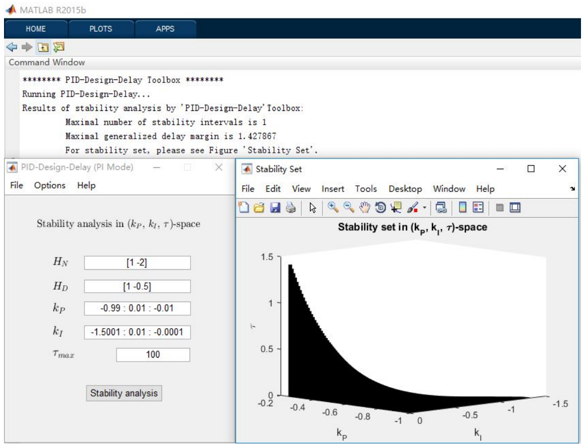


Fig. 5. Application of *PID-Design-Delay* toolbox to Example 1

Example 2: The consumed time by the toolbox depends on the task complexity. We here consider three cases for k_P and k_I inputs for the system of Example 1.

Case 1: Input "-0.99:0.01:-0.01" in k_P edit box and "-1.5001:0.01:-0.0001" in k_I edit box;

Case 2: Input "-0.99:0.005:-0.01" in k_P edit box and "-1.5001:0.005:-0.0001" in k_I edit box;

Case 3: Input "-0.99:0.001:-0.01" in k_P edit box and "-1.5001:0.001:-0.0001" in k_I edit box.

Case 1, Case 2, and Case 3 (the ranges for k_P and k_I are the same, but the step lengths are different) correspond to different complexity. The consumed time in the three cases (on a PC with an Intel Core 3.40GHz CPU with 32G RAM) is given in Table I. It is indicated that the shorter the step lengths are, the longer the consumed time will be. □

TABLE I
COMPUTATION TIME (SECONDS)

	Case 1	Case 2	Case 3
Computation time	5.340897	21.004788	518.107136

C. Stability analysis in PD mode

If 'PD controller' is selected, the dialog box as shown in Fig. 3(b) will appear.

The PD controller is described by the transfer function

$$k_P + k_D \lambda, \tag{10}$$

and the characteristic function of the closed-loop system is

$$f(\lambda,\tau) = H_{\mathrm{D}}(\lambda) + H_{\mathrm{N}}(\lambda)(k_P + k_D\lambda)e^{-\tau\lambda}. \tag{11}$$

There are two cases that the PD controller definitely cannot stabilize.

In the case $\deg(H_{\mathrm{D}}(\lambda)) = \deg(H_{\mathrm{N}}(\lambda))$, the closed-loop system is of the advanced type if the PD controller is used. Thus, the closed-loop system definitely cannot be asymptotically stable. Accordingly, some indications will be given in the command window.

In the case $\deg(H_{\mathrm{D}}(\lambda)) = \deg(H_{\mathrm{N}}(\lambda))+1$, the closed-loop system is of the neutral type if the PD controller is used. A necessary condition for the system stability is

$$|k_D| < |\frac{a_{\mathrm{n}}}{b_{\mathrm{m}}}|. \tag{12}$$

If the above necessary condition is violated, some indications will be displayed in the command window accordingly.

The operation in the PD mode is similar to that in the PI mode.

Example 3: Apply the PD controller to the following controlled plant

$$\frac{1}{\lambda^2 - 1.4\lambda + 0.48}.$$

Suppose the stability in the domain $(k_P, k_D, \tau) \in [-0.5, 0.5] \times [-1.5, 1.5] \times [0, 100]$ is of interest, and the step lengths for k_P and k_D are both to be set as 0.001.

The user should fill H_{N} edit box with "[1]", H_{D} edit box with "[1 -1.4 0.48]", k_P edit box with "-0.5:0.001:0.5", k_D edit box with "-1.5:0.001:1.5", and τ_{max} edit box with "100".

After clicking 'Stability analysis' button, the toolbox will automatically perform the analysis. Then, the stability results will be displayed in the command window as well as in 'Stability Set' figure. The application of *PID-Design-Delay* toolbox to Example 3 is shown in Fig. 6. □

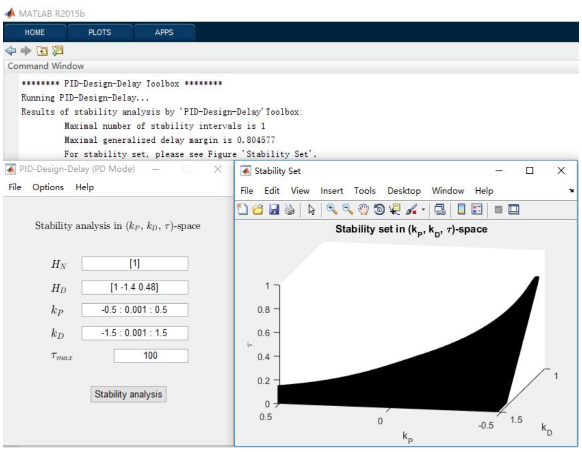


Fig. 6. Application of *PID-Design-Delay* toolbox to Example 3

D. Stability analysis in PID mode

If 'PID controller' is selected, the dialog box as shown in Fig. 3(c) will appear.

The cases that the PID controller definitely cannot stabilize are the same as those for the PD controller.

The operation in the PID mode is similar to that in the PI mode. Please note that the color information in 'Stability Set' figure stands for the generalized delay margin.

Example 4: Apply the PID controller to the following controlled plant

$$\frac{1}{\lambda^2 - \frac{2}{\pi}\lambda + 1 + \frac{2}{\pi}}.$$

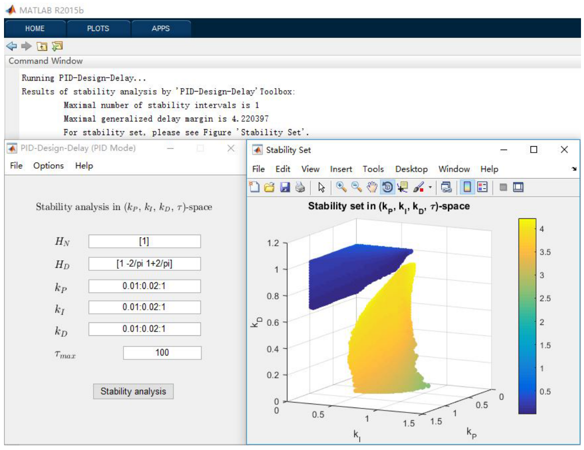


Fig. 7. Application of *PID-Design-Delay* toolbox to Example 4

Suppose that the stability in the domain $(k_P, k_I, k_D, \tau) \in [0.01, 1] \times [0.01, 1] \times [0.01, 1] \times [0, 100]$ is of interest, and that the step lengths for k_P, k_I, and k_D are all to be set as 0.01.

The user should fill H_{N} edit box with "[1]", H_{D} edit box with "[1 -2/pi 1+2/pi]", k_P edit box with "0.01:0.01:1", k_I edit box with "0.01:0.01:1", k_D edit box with "0.01:0.01:1", and τ_{max} edit box with "100".

After clicking 'Stability analysis' button, the toolbox will automatically perform the analysis. Then, the stability results will be displayed in the command window as well as in 'Stability Set' figure. The application of *PID-Design-Delay* toolbox to Example 4 is shown in Fig. 7. □

IV. Menu bars

The functions of menu bars in PI, PD, and PID modes are quite similar. To save space, we only introduce the menu bar in PI mode, in the sequel.

The menu bar has three menus: File, Options, and Help (see Fig. 3(a)).

File menu:

The drop-down list of File menu is shown in Fig. 8, which contains four topics: Open, Save, Clear, and Return.

Open: Open an existing MAT-file, which saves a set of inputs for the dialog box shown in Fig. 3(a).

Save: Save all the inputs in current dialog box to a MAT-file.

Clear: Clear all the inputs in current dialog box.
Return: Return to the previous dialog box (i.e., the dialog box shown in Fig. 2).

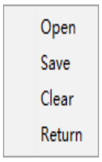
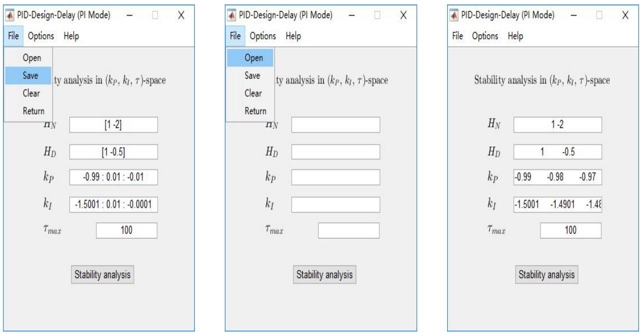


Fig. 8. File drop-down menu list

Example 5: We here explain how to facilitate the implementation (if some information is used more than once), with the aid of 'Save' and 'Open' buttons. Consider again the task in Example 1.

After entering all the necessary information in the dialog box, we may save all the inputs (Fig. 9(a)) into a MAT-file, say Example1.mat (Fig. 10).



(a) Saving information (b) Opening MAT-file (c) Invoked information

Fig. 9. Using 'Save' and 'Open' buttons for Example 5

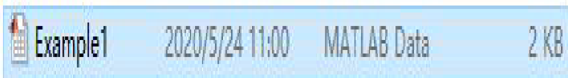


Fig. 10. Saved MAT-file for Example 5

Later, if we want to invoke the inputs, press 'Open' button (Fig. 9(b)) and select the file shown in Fig. 10. As a result, the information inputted earlier can be invoked (Fig. 9(c)). □

Options:

When 'Options' is selected, the dialog box as shown in Fig. 11 pops up. The dialog box consists of three types of boxes: five check boxes, an edit box, and a list box.

Five check boxes are as follows.

'maximal number of stability delay-intervals' check box;

'maximal delay margin' check box;

'(k_P, k_I) at which the maximal delay margin is found' check box;

'maximal generalized delay margin' check box;

'(k_P, k_I) at which the maximal generalized delay margin is found' check box.

If a check box is selected (indicated by '$\surd$'), the corresponding result will be displayed in the command window when the stability analysis is finished.

'$\tilde{\sigma}$' edit box: Set a very small positive value for $\tilde{\sigma}$.

Define $\sigma \triangleq \{\min|\mathrm{Re}(\lambda)| : f(\lambda,\tau) = 0\}$. The system is considered to be in or close to the state "having CIRs when $\tau = 0$" in the domain $\sigma \le \tilde{\sigma}$.

'size of points' list box: Set the size of points for 'Stability set' figure.

'Apply' button: Activate the modified options.

'Default' button: Restore the default options.

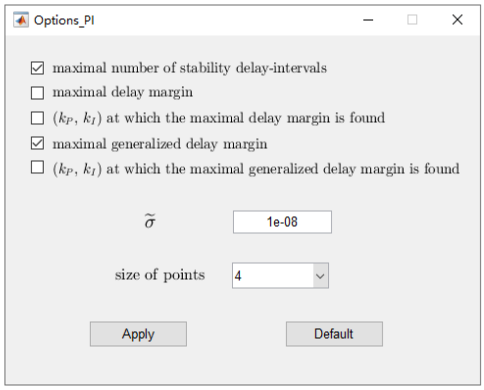


Fig. 11. Dialog box for options in PI mode

The default options are as shown in Fig. 11. The stability results of Example 1 (Fig. 5) are based on the default options.

Help:

When 'Help' is selected, a brief introduction to the PI mode pops up.

V. CASE STUDY

In this section, we consider a speed-controlled DC motor drive system as depicted in Fig. 12, where $n(t)$ is the speed of DC motor and U_n^* is the speed reference voltage. There are four major components: a (separately-excited) DC motor, a (thyristor) converter, a tachogenerator, and a PI controller.

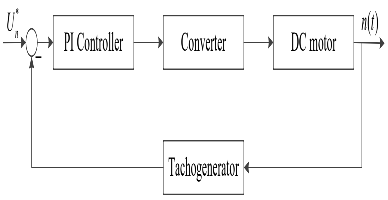


Fig. 12. Block diagram of DC motor drive system

The transfer function of the DC motor is in the form

$$\frac{\mathrm{b}_0}{\mathrm{a}_2\lambda^2 + \mathrm{a}_1\lambda^2 + 1}, \tag{13}$$

where $\mathrm{a}_2, \mathrm{a}_1, \mathrm{b}_0$ are real numbers.

The thyristor converter can be modelled as

$$\mathrm{c}_0 e^{-\tau\lambda}, \tag{14}$$

where c_0 is the gain and τ is the phase delay.

The DC motor is equipped with a tachogenerator, which produces a DC voltage serving as the speed feedback. The tachogenerator can be modelled as a gain d_0.

With the DC motor drive system, we can control the speed of DC motor $n(t)$ (the desired speed is $\frac{U_n^*}{d_0}$). It is seen that $n(t) \to \frac{U_n^*}{d_0}$ if and only if the closed-loop system is asymptotically stable.

The characteristic function for DC motor drive system is

$$f(\lambda, \tau) = a_2\lambda^3 + a_1\lambda^2 + \lambda + b_0c_0d_0(k_P\lambda + k_I)e^{-\tau\lambda}. \quad (15)$$

Comparing the characteristic functions (8) and (15) , we can see that the stability of DC motor system is equivalent to that the closed-loop system as shown in Fig. 1 with the following controlled plant

$$H_0(\lambda) = \frac{b_0c_0d_0}{a_2\lambda^2 + a_1\lambda + 1}. \quad (16)$$

Example 6: Choose the associated coefficients as in Example 3.6 of [6], and the corresponding transfer function (16) is

$$\frac{1.3139816625}{0.00224016\lambda^2 + 0.1285\lambda + 1}.$$

We let the controller gains (k_P and k_I) and the delay τ be all free parameters. In view of (17), we enter "[1.3139816625]" in H_N edit box and "[0.00224016 0.1285 1]" in H_D edit box. In addition, we enter "0.1:0.0001:1" in k_P edit box, "0.1:0.01:1" in k_I edit box, and "100" in τ_{max} box.

After clicking 'Stability analysis' button, the toolbox will automatically perform the analysis. Then, the stability results will be displayed in the command window as well as in 'Stability Set' figure. The application of *PID-Design-Delay* toolbox to Example 6 is shown in Fig. 13.

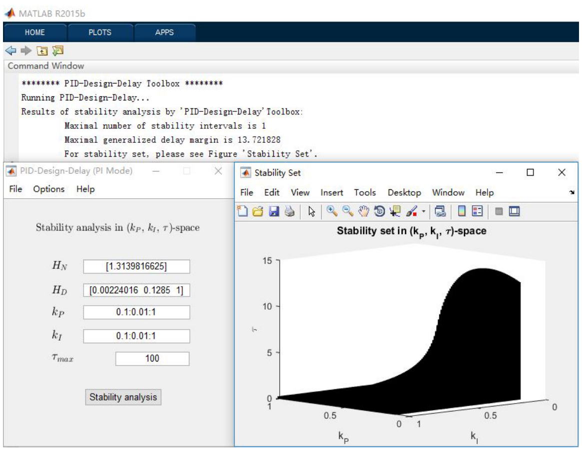


Fig. 13. Application of *PID-Design-Delay* toolbox to Example 6

Based on the above analysis, the speed of DC motor $n(t)$ converges to $\frac{U_n^*}{d_0}$, if and only if (k_P, k_I, τ) lies in the stability set shown in 'Stability Set' figure as shown in Fig. 13. □

VI. Conclusion

In this paper, we introduced a MATLAB-based toolbox, called PID-Design-Delay, for the PID stabilization of time-delay systems, where the three controller gains (k_P , k_I , and k_D) and the delay (τ) are all treated as free parameters. The toolbox can automatically fulfill the stability analysis and find the whole stability set in the parameter space of (k_P, k_I, k_D, τ).

PID-Design-Delay has been used in teaching of undergraduate students in Northeastern University, China. It turns out: With the manual and demo videos, a undergraduate majoring in automation can learn to design the PID controller and find the whole stability set by using this toolbox, within two hours.

Acknowledgment

This research was supported in part by National Natural Science Foundation of China under Grant 61733003.

References

[1] K. J. Åström and T. Hägglund, *PID Controllers: Theory, Design, and Tuning.* Research Triangle Park, NC, Instrument Society of America, 1995.

[2] D. Breda, S. Maset, and R. Vermiglio, *TRACE-DDE: a tool for robust analysis and characteristic equations for delay differential equations.* In Loiseau, J. J. et al. (Eds.), Topics in Time Delay Systems: Analysis, Algorithm and Control, Berlin Heidelberg, Springer, 2009, pp. 145-155.

[3] K. Engelborghs, T. Luzyanina, and D. Roose, "Numerical bifurcation analysis of delay differential equations using DDE-BIFTOOL," *ACM Trans. Math. Software,* vol. 28, pp. 1-21, 2002.

[4] K. Gu, V. L. Kharitonov, and J. Chen, *Stability of Time-Delay Systems.* Boston, Birkhäuser, 2003.

[5] P. Ju and H. Zhang, "Further results on the achievable delay margin using LTI control," *IEEE Trans. Autom. Control,* vol. 61, pp. 3134-3139, 2016.

[6] R. Krishnan, *Electric Motor Drives: Modeling, Analysis, and Control.* Prentice Hall, Upper Saddle River, New Jersey, 2001.

[7] M. S. Lee and C. S. Hsu, "On the τ-decomposition method of stability analysis for retarded dynamical systems," *SIAM J. Control,* vol. 7, pp. 242-259, 1969.

[8] X.-G. Li, S.-I. Niculescu, and A. Çela, *Analytic Curve Frequency-Sweeping Stability Tests for Systems with Commensurate Delays.* London, Springer, 2015.

[9] X.-G. Li, S.-I. Niculescu, J.-X. Chen, and T. Chai, "Characterizing PID controllers for linear time-delay systems: A parameter-space approach," Provisionally accepted by *IEEE Trans. Autom. Control*.

[10] X.-G. Li, S.-I. Niculescu, A. Çela, L. Zhang, and X. Li, "A frequency-sweeping framework for stability analysis of time-delay systems," *IEEE Trans. Autom. Control,* vol. 62, pp. 3701-3716, 2017.

[11] D. Ma and J. Chen, "Delay margin of low-order systems achievable by PID controllers," *IEEE Trans. Autom. Control,* vol. 64, pp. 1958-1973, 2019.

[12] R. H. Middleton and D. E. Miller, "On the achievable delay margin using LTI control for unstable plants," *IEEE Trans. Autom. Control,* vol. 52, pp. 1194-1207, 2007.

[13] W. Michiels and S.-I. Niculescu, *Stability, Control, and Computation for Time-Delay Systems: An Eigenvalue-Based Approach.* Philadelphia, SIAM, 2014.

[14] J. Neimark, "D-subdivisions and spaces of quasi-polynomials," *Prikl. Mat. Meh.,* vol. 13, pp. 349-380, 1949.

[15] T. Qi, J. Zhu, and J. Chen, "Fundamental limits on uncertain delays: When is a delay system stabilizable by LTI controllers?" *IEEE Trans. Autom. Control,* vol. 62, pp. 1314-1328, 2017.

[16] G. J. Silva, A. Datta, and S. P. Bhattacharyya, *PID Controllers for Time-Delay Systems.* Boston, Birkhäuser, 2005.

[17] T. Vyhlídal, and P. Zítek, "Mapping based algorithm for large-scale computation of quasipolynomial zeros," *IEEE Trans. Autom. Control,* vol. 54, pp. 171-177, 2009.