HAL
open science

New Features of P3δ software: Partial Pole Placement via Delay Action

Islam Boussaada, Guilherme Mazanti, Silviu-Iulian Niculescu, Adrien Leclerc, Jayvir Raj, Max Perraudin

▶ **To cite this version:**

Islam Boussaada, Guilherme Mazanti, Silviu-Iulian Niculescu, Adrien Leclerc, Jayvir Raj, et al.. New Features of P3δ software: Partial Pole Placement via Delay Action. TDS 2021 - 16th IFAC Workshop on Time Delay Systems, Sep 2021, Guangzhou, China. pp.215-221, 10.1016/j.ifacol.2021.11.142 . hal-03298710v2

HAL Id: hal-03298710

https://hal.science/hal-03298710v2

Submitted on 17 Aug 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

New Features of P3δ software: Partial Pole Placement via Delay Action

Islam Boussaada [*,**] **Guilherme Mazanti** [*]
Silviu-Iulian Niculescu [*] **Adrien Leclerc** [**] **Jayvir Raj** [**]
Max Perraudin [**]

[*] *Université Paris-Saclay, CNRS, CentraleSupélec, Inria, Laboratoire des Signaux et Systèmes, 91190, Gif-sur-Yvette, France. (e-mail: firstname.lastname@l2s.centralesupelec.fr).*
[**] *IPSA, 63 Boulevard de Brandebourg, 94200 Ivry-sur-Seine, France*

Abstract: This paper presents the software *Partial Pole Placement via Delay Action*, or P3δ for short. P3δ is a Python software with a friendly user interface for the design of parametric stabilizing feedback laws with time-delays, thanks to two properties of the distribution of quasipolynomials' zeros, called multiplicity-induced-dominancy and coexisting real roots-induced-dominancy. After recalling recent theoretical results on these properties and their use for the feedback stabilization of control systems operating under time delays, the paper presents the main features of the current version of P3δ. We detail, in particular, the assignable admissible region (the set of allowable dominant roots and the corresponding delay), which helps the user in the choice of input information, allowing a reliable stabilizing delayed feedback. We also present the newly set online version of P3δ.

Keywords: Delay, Stability, Controller design, Python toolbox, GUI, Online software.

1. INTRODUCTION

Even though time delays commonly lead to desynchronizing or destabilizing effects on the dynamics of the system they appear, some works have emphasized that delays may also have stabilizing effects in control design. One of the earliest such works is Tallman and Smith (1958), where authors use a delay-based controller to improve the stability of systems with oscillatory behavior and small damping. Another strategy, used for instance in Suh and Bien (1979) and Atay (1999), is to replace the classical proportional-derivative controller by a proportional-delayed controller, using a so-called "average derivative action" due to the delay. It should also be noted that closed-loop stability may be guaranteed for some control systems precisely by the existence of the delay, a fact highlighted in particular in Niculescu et al. (2010), in which the reader may find further discussions on the stabilizing effects of delays. A growing literature exhibits the design of delayed controllers in a wide range of applications, such as the control of flexible mechanical structures or the regulation of networks (see, e.g., Boussaada et al. (2018); Irofti et al. (2016)).

In this paper, we present a software, P3δ (which stands for *Partial Pole Placement via Delay Action*), which helps its user in the stability analysis and the stabilization of linear time-invariant delay-differential equations of retarded or neutral type with a single time delay, under the form

$$y^{(n)}(t) + a_{n-1}y^{(n-1)}(t) + \cdots + a_0 y(t) + b_m y^{(m)}(t-\tau) + \cdots + b_0 y(t-\tau) = 0, \quad (1)$$

where $\tau > 0$ is the positive delay, y is the real-valued unknown function, n and m are nonnegative integers with $n \geq m$, and $a_0, \ldots, a_{n-1}, b_0, \ldots, b_m$ are real coefficients. Equation (1) is said to be of *retarded type* if $n > m$, that is, the highest-order derivative only appears in the non-delayed term $y^{(n)}(t)$, and it is said to be of *neutral type* if $m = n$ and $b_n \neq 0$, which corresponds to a highest-order derivative appearing in both non-delayed term $y^{(n)}(t)$ and delayed one $y^{(n)}(t-\tau)$.

A typical situation in which equations under the form (1) arise is the delayed feedback stabilization of a linear time-invariant controlled differential equation of the form $y^{(n)} + a_{n-1}y^{(n-1)}(t) + \cdots + a_0 y(t) = u(t)$, when applying a delayed feedback control $u(t) = -b_m y^{(m)}(t-\tau) - \cdots - b_0 y(t-\tau)$. In that case, the choice of the free parameters $b_0, \ldots, b_m$ in the feedback control will have an important influence on the behavior of the closed-loop system. We refer the reader to Boussaada and Niculescu (2018); Boussaada et al. (2020b, 2018); Mazanti et al. (2021a,b) for more examples of systems under the form (1) to which the methods of the present paper can be applied.

A classical technique to address the stability analysis of linear time-invariant time-delay systems is by the use of spectral methods (see, e.g., Hale and Verduyn Lunel (1993); Michiels and Niculescu (2014)), which consist on the study of the complex roots of a *characteristic function* of the system. The characteristic function of (1) is

$$\Delta(s) = s^n + \textstyle\sum_{k=0}^{n-1} a_k s^k + e^{-s\tau} \sum_{k=0}^{m} b_k s^k, \quad (2)$$

and (1) is exponentially stable if and only if the *spectral abscissa* $\gamma = \sup\{\operatorname{Re} s \mid \Delta(s) = 0\}$ satisfies $\gamma < 0$.

Characteristic functions of time-delay systems are *quasi-polynomials*, i.e., functions which can be written as a finite sum of polynomials multiplied by exponentials. Due

to their applications in the spectral analysis of time-delay systems, the study of quasipolynomials has been the subject of several works, such as Berenstein and Gay (1995); Hale and Verduyn Lunel (1993); Stépán (1989); Wielonsky (2001). Except for the particular case where no exponentials are present and the quasipolynomial reduces to a polynomial, quasipolynomials have infinitely many roots in the complex plane. In the context of stabilization of a control system by a delayed feedback law, one only disposes of finitely many parameters in the feedback law to choose the location of these infinitely many roots and place them in order to guarantee a negative spectral abscissa, and hence exponential stability of the closed-loop system.

A possible strategy to stabilize a time-delay system is to select the free parameters of the system in order to choose the location of finitely many roots while also guaranteeing that the *dominant root*, i.e., the rightmost root on the complex plane, is among the chosen ones. This has been the subject of several recent works, such as Amrane et al. (2018); Bedouhene et al. (2020); Boussaada and Niculescu (2018); Boussaada et al. (2020b, 2018); Mazanti et al. (2021a, 2020a,b). Contrarily to the strategy of finite spectrum assignment used, e.g., in Manitius and Olbrot (1979), the controllers designed using these techniques do not render the closed-loop system finite-dimensional, but control instead its rightmost spectral value. These methods also extend to some partial differential equations, as detailed, for instance, in Mazanti et al. (2021b).

Notice that there exist other pole placement paradigms for time-delay systems, such as the *continuous pole-placement* introduced in Michiels et al. (2002). Based on the continuous dependence of the characteristic roots on the controller parameters, this technique consists in shifting the unstable characteristic roots from $\mathbb{C}_+$ to $\mathbb{C}_-$ in a "quasi-continuous" way, subject to the constraint that, during this shifting action, stable characteristic roots are not crossing the imaginary axis from $\mathbb{C}_-$ to $\mathbb{C}_+$ (see also Ram et al. (2011)).

Two main strategies have been used in the works Amrane et al. (2018); Bedouhene et al. (2020); Boussaada and Niculescu (2018); Boussaada et al. (2020b, 2018); Mazanti et al. (2021a, 2020a,b) to assign finitely many roots while guaranteeing that the rightmost root is among them. The first one consists in assigning a real root of maximal multiplicity and proving that this root is necessarily the rightmost root of the characteristic quasipolynomial, a property which has been named *multiplicity-induced-dominancy*, or MID for short. The second strategy consists in assigning a certain amount of real roots, typically equally spaced for simplicity, and proving that the rightmost root among them is also the rightmost root of the characteristic quasipolynomial, a property which has been named *coexisting real roots-induced-dominancy*, or CRRID for short.

The MID property for (1) was shown, for instance, in Boussaada et al. (2018) in the case $n = 2$ and $m = 0$, in Boussaada et al. (2020b) in the case $n = 2$ and $m = 1$ (see also Boussaada and Niculescu (2018)), and in Mazanti et al. (2021a) in the case of any positive integer n and $m = n - 1$ (see also Mazanti et al. (2020a)). It was also studied for neutral systems of orders 1 and 2 in Ma et al. (2020); Benarab et al. (2020); Mazanti et al. (2021b), and extended to complex conjugate roots of maximal multiplicity in Mazanti et al. (2020b). The CRRID property was shown, for instance, in Amrane et al. (2018) in the cases $(n, m) = (2, 0)$ and $(n, m) = (1, 0)$, and in Bedouhene et al. (2020) in the case of any positive integer n and $m = 0$.

In all the above cases, the maximal multiplicity of a real root or, equivalently, the maximal number of coexisting simple real roots is the integer $n + m + 1$. Furthermore, the idea to exploit the nature of (real or complex) open-loop roots in control design was proposed for second-order systems in Boussaada et al. (2020b) and further extended for arbitrary order systems with real-rooted plants in Balogh et al. (2020, 2021).

Based on the results from Amrane et al. (2018); Bedouhene et al. (2020); Boussaada and Niculescu (2018); Boussaada et al. (2020b, 2018); Mazanti et al. (2021a, 2020a,b), the Python software P3δ allows for the parametric design of stabilizing feedback laws with time delays using the MID and CRRID properties. It has the advantage to allow constructive methods more appropriate for understanding the effect of uncertainties on the spectrum distribution. The first version of P3δ was formerly described in Boussaada et al. (2020a) and it covered the design of feedback laws for linear time-invariant differential equations with a single time delay under the form (1) using MID techniques in retarded case. The newer version of P3δ, described in the present paper, benefits also from the CRRID property and treats both retarded and neutral equations.

Stability, robustness, or bifurcation aspects of time-delay systems have also been the subject of other recently developed softwares, such as YALTA (Avanessoff et al., 2014), which performs H_∞ stability analysis of time-delay systems with commensurate delays, TRACE-DDE (Breda et al., 2009), for the computation of characteristic roots and stability charts of linear autonomous time-delay systems, DDE-BIFTOOL (Engelborghs et al., 2002), interested in the computation, continuation, and stability analysis of steady-state solutions of time-delay systems and their bifurcations, and QPmR (Vyhlídal and Zítek, 2014), for the computation of roots of quasipolynomials. One of the major novelties of P3δ lies in addressing the stabilization of control systems with time delays by using the MID and CRRID properties to design stabilizing feedback laws, making use of both symbolic and numeric computations.

2. DESCRIPTION OF P3δ

P3δ is freely available for download on `https://cutt.ly/p3delta`, where installation instructions, video demonstrations, and the user guide are also available. Interested readers may also contact directly any of the authors of the paper.

Three modes are implemented in the current version of P3δ: "Generic MID", "Generic CRRID", and "Control-oriented MID". In the generic modes, all coefficients of (1) are assumed to be available for choice, whereas, in the control-oriented mode, P3δ assumes that $a_0, \ldots, a_{n-1}$ are fixed and only $b_0, \ldots, b_m$ are free. These three modes are detailed in Sections 2.1, 2.2, and 2.3 below.

2.1 Generic MID mode

The "Generic MID" mode was already implemented in the first version of P3δ and is described in Boussaada et al. (2020a). For sake of completeness, we also provide a description of this mode here.

In the "Generic MID" mode, the user inputs the values of the delay τ and of the desired real root s_0 and P3δ computes all coefficients $a_0, \ldots, a_{n-1}, b_0, \ldots, b_m$ ensuring that the value s_0 is a dominant root of Δ of maximal multiplicity $n+m+1$, using the procedure from Mazanti et al. (2021a); Boussaada et al. (2021). To use the "Generic MID" mode, the user should proceed as follows:

1. Enter the values of the integers n and m appearing in the differential equation (1).
2. Select the "Generic MID" option in the drop-down menu "— Choose type —".

After this selection, the window of the program is filled with places for the other inputs and outputs of P3δ.

3. Enter the values of the desired real root of maximal multiplicity s_0 and of the delay τ in the corresponding fields that appear below the drop-down menu.
4. Enter the bounds $x_{\min}, x_{\max}, y_{\min}, y_{\max}$ of the rectangle $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}] \subset \mathbb{C}$ in which P3δ will look for roots of (2) and press the "Confirm" button.

Once the "Confirm" button is pressed, P3δ computes and displays the values of the coefficients $a_0, \ldots, a_{n-1}, b_0, \ldots, b_m$ ensuring that s_0 is a root of maximal multiplicity of the quasipolynomial Δ from (2), performs a numerical computation of all roots of Δ within the selected rectangle, and plot these roots in the plot "Roots" at the lower left corner of the window. The latter computation is carried out using Python's `cxroots` module (see Parini (2018–)).

Optionally, after the previous computations are completed, the user may also simulate some trajectories of the system in time domain. This can be done, after completing step 4 above, by the following steps:

5. Choose the type of the initial condition from the drop-down menu "— Initial Solution —".

The currently supported types are "Constant", "Polynomial", "Exponential", and "Trigonometric", which corresponds to initial conditions of the forms $x(t) = c$, $x(t) = \sum_{k=0}^{r} c_k t^k$, $x(t) = Ae^{\gamma t}$, and $x(t) = A\sin(\omega t + \varphi)$, respectively, where $c, r, c_0, \ldots, c_r, A, \gamma, \omega, \varphi$ are constants to be chosen by the user and the initial condition is defined in the time interval $[-\tau, 0]$.

6. Enter the simulation time T in the corresponding box.
7. Enter the values of the constants appearing in the expression of the initial condition in the corresponding input boxes.
8. After entering all the constants, press "Enter" on the keyboard or click on the "Confirm" button appearing in the same frame as the constants.

After these steps, the numerical solution corresponding to the chosen initial condition is computed using an explicit Euler scheme in the time interval $[-\tau, T]$ and plotted in the graph on the "Solutions" part of the screen.

2.2 Generic CRRID mode

In the newly implemented "Generic CRRID" mode, the user inputs the values of the delay τ and of $n+m+1$ desired real roots $s_1 \geq \cdots \geq s_{m+n+1}$ and P3δ computes all coefficients $a_0, \ldots, a_{n-1}, b_0, \ldots, b_m$ ensuring that the value s_1 is simple and dominant root of Δ, using the procedure described in Bedouhene et al. (2020).

The use of the "Generic CRRID" mode is very similar to that of the "Generic MID" mode described in Section 2.1, with the differences that, in Step 2, the user should select the "Generic CRRID" mode and, in Step 3, the user should enter the values of the desired real roots $s_1 \geq \cdots \geq s_{m+n+1}$ and of the delay τ in the corresponding fields.

As in the "Generic MID" mode, once the "Confirm" button is pressed, P3δ will compute and show the values of the coefficients $a_0, \ldots, a_{n-1}, b_0, \ldots, b_m$ ensuring that the real roots $s_1 \geq \cdots \geq s_{m+n+1}$ are simple roots of the quasipolynomial Δ from (2), and numerically compute all roots of Δ within the selected rectangle using Python's `cxroots` module. The user can simulate trajectories in time domain by following Steps 5–8 from Section 2.1.

2.3 Control-oriented MID mode

Contrarily to the previous two modes, the "Control-oriented MID" mode considers that the coefficients $a_0, \ldots, a_{n-1}$ corresponding to the non-delayed terms of (1) are given and that the coefficients $b_0, \ldots, b_m$ corresponding to the delayed terms are available for choice. As described in Boussaada et al. (2020b,a), in this mode, one imposes the existence of a real root of multiplicity $m+2$ at s_0 by requiring that $\Delta(s_0) = \Delta'(s_0) = \cdots = \Delta^{(m+1)}(s_0) = 0$. This gives a system of $m+2$ equations on the $m+1$ unknowns $b_0, \ldots, b_m$, and thus another parameter of the system, either the delay τ or the root s_0, must be considered as an unknown of the problem, while the value of the other is assumed to be fixed. Hence, in "Control-oriented MID" mode of P3δ, the user must choose to input either the value of τ or that of s_0 (but not both) and P3δ computes all coefficients $b_0, \ldots, b_m$ ensuring the existence of a dominant root of the quasipolynomial Δ from (2) of multiplicity $m+2$, as well as the value of the parameter τ or s_0 that has not been fixed by the user.

To use the "Control-oriented MID" mode, the user should proceed as in Section 2.1, but selecting "Control-oriented MID" in Step 2 and then entering either s_0 or τ, as well as the values of the known coefficients $a_0, \ldots, a_{n-1}$. As in the previous two modes, the user may also select to perform a time-domain simulation by following the same steps.

Remark 1. In the "Control-oriented MID" mode, it may happen to be impossible to choose a real root s_0 of multiplicity $m+2$. In this case, the previous version of P3δ warned the user of this fact and provided an equation relating s_0 and τ. The user should either enter a value of s_0 such that this equation admits a positive root τ or a positive value of τ such that this equation admits a real root s_0 in order to proceed with the computations. Instead of providing this equation, the new version of P3δ provides the plot in the plane (s_0, τ) of the points for which the equation is satisfied. This region of admissible assignment

should help the user in the choice of an allowable s_0 or τ. This new feature is further described in Section 2.4.

In the case where the user inputs the delay τ instead of the desired root s_0, the "Control-oriented MID" mode can also perform a numerical sensitivity analysis of the computed roots with respect to variations of τ. The steps to get the sensitivity plot are the following:

1. Select the "Sensitivity" tab in the "Roots" plot.
2. Select "tau sensitivity" in the drop-down menu "— Sensitivity —" above the "Roots" plot.
3. Enter the value of the step ε and the number of iterations K in the corresponding boxes.
4. Enter the bounds $x_{\min}, x_{\max}, y_{\min}, y_{\max}$ of the rectangle $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}] \subset \mathbb{C}$ in which P3δ will look for roots of (2).

Since the sensitivity computation may take quite some time, it is highly recommended to choose a smaller rectangle containing only a few roots of Δ.

5. Press the "Confirm" button in the frame of the bounds of the rectangle.

The sensitivity plot appears in the "Roots" plot and contains the roots of Δ in the selected rectangle for the values of delays $\tau + k\varepsilon$ for $k \in \{-K, -K+1, \ldots, K-1, K\}$. Roots computed with negative values of k, corresponding to values of the delay smaller than τ, are represented in shades of blue, with darker blue representing $k = -K$ and lighter tones representing increasing values of k. Roots computed with positive values of k, corresponding to values of the delay larger than τ, are represented in shades of orange to red, with darker red representing $k = K$ and lighter tones moving to orange representing decreasing values of k. The roots computed with $k = 0$, corresponding to the nominal value of τ selected by the user, are represented by black diamonds.

2.4 Assignment admissibility region

This section describes the admissibility region and how to display it using P3δ. Note that this new feature of P3δ is available only in the "Control-Oriented MID" mode, since there are no constraints on (s_0, τ) in the other modes.

For given coefficients $a_0, \ldots, a_{n-1}$, the admissibility region is defined as the set of pairs $(s_0, \tau) \in \mathbb{R} \times (0, +\infty)$ for which there exist real coefficients $b_0, \ldots, b_m$ such that s_0 is a root of Δ of multiplicity at least $m+2$ when the delay is τ.

Let us first describe how this admissibility region can be determined. Consider the quasipolynomial Δ from (2) with known values for the coefficients $(a_i)_{0 \le i \le n-1}$. We impose that $s_0 \in \mathbb{R}$ is a root of multiplicity at least $m+2$ of Δ. We use the $m+1$ equations $\Delta^{(k)}(s_0) = 0$, $k \in \{0, \ldots, m\}$, in order to express the coefficients $(b_i)_{0 \le i \le m}$ in terms of $(a_i)_{0 \le i \le n-1}$, s_0, and τ. This is always possible, since these $m+1$ equations are linear in the $m+1$ variables $b_0, \ldots, b_m$ and this linear system can be shown to admit a unique solution.

In order for s_0 to be a root of multiplicity $m+2$ of Δ, in addition to the $m+1$ equations $\Delta^{(k)}(s_0) = 0$, $k \in \{0, \ldots, m\}$, the $m+1$-th derivative of Δ must also be zero at s_0, i.e., the equation $\Delta^{(m+1)}(s_0) = 0$ must also be satisfied. By replacing the previously found expressions of $(b_i)_{0 \le i \le m}$ into the equation $\Delta^{(m+1)}(s_0) = 0$, one obtains a relation between s_0, τ, and the coefficients $(a_i)_{0 \le i \le n-1}$, which is a necessary and sufficient condition for s_0 to be a root of multiplicity at least $m+2$ of Δ. Since the coefficients $(a_i)_{0 \le i \le n-1}$ are assumed to be known, the only unknowns in this equation are s_0 and τ, and hence the admissibility region is the set of pairs $(s_0, \tau) \in \mathbb{R} \times (0, +\infty)$ satisfying this equation.

These computations are implemented symbolically in P3δ using Python's `sympy` module and, once the equation describing the admissibility region is obtained, the admissibility region is displayed to the user. Since it is not possible to display the full admissibility region in $\mathbb{R} \times (0, +\infty)$, the user is prompted for values $s_{0,\min} < 0$ and $\tau_{\max} > 0$ and only the part of the admissibility region inside the rectangle $[s_{0,\min}, 0] \times [0, \tau_{\max}]$ is displayed.

The admissibility region is always displayed when the user chooses the "Control-oriented MID" mode of P3δ, after performing the following steps:

1. Enter the values of the integers n and m appearing in the differential equation (1).
2. Select the "Control-oriented MID" option in the drop-down menu "— Choose type —".

After this selection, the window of the program is filled with places for the other inputs and outputs of P3δ.

3. Enter the values of the known coefficients $a_0, \ldots, a_{n-1}$.
4. Enter the values of the limits $\tau_{\max} > 0$ and $s_{0,\min} < 0$ of the rectangle $[s_{0,\min}, 0] \times [0, \tau_{\max}]$ in which to plot the admissibility region in the respective fields "tau limit" and "s0 limit".
5. Click on "Confirm".

A new window appears with the admissibility region plotted. As an example, Figure 1 shows the admissibility region in the case $n = 2$, $m = 1$, $a_0 = a_1 = 1$, and with $\tau_{\max} = 3$ and $s_{0,\min} = -10$.

Fig. 1. Admissibility plot for a second-order retarded system with $a_0 = 1$, $a_1 = 1$, $\tau_{\max} = 3$, and $s_{0,\min} = -10$.

After the admissibility region is shown, the user is prompted to enter a value for either s_0 or τ in order to proceed with the "Control-oriented MID" mode, as

described in Section 2.3. The value of s_0 or τ entered by the user should be such that a point with the corresponding value of s_0 or τ exists in the admissibility region. In the example of Figure 1, the user cannot enter a value of τ exceeding ≈ 1.6, nor a value of s_0 exceeding ≈ -1.5.

3. ILLUSTRATIVE EXAMPLE: STABILIZATION OF AN OSCILLATOR

As an illustration of the use of P3δ, let us consider the stabilization of an oscillator described by the equation $y''(t) + \omega^2 y(t) = u(t)$ with a delayed feedback control $u(t) = -b_0 y(t-\tau)$, which yields the delay-differential equation

$$y''(t) + \omega^2 y(t) + b_0 y(t-\tau) = 0, \tag{3}$$

whose characteristic quasipolynomial is $\Delta(s) = s^2 + \omega^2 + b_0 e^{-s\tau}$. This corresponds to (1) with $n = 2$, $m = 0$, $a_1 = 0$, and $a_0 = \omega^2$. Immediate computations show that $s_0 \in \mathbb{R}$ is a root of multiplicity at least $m + 2 = 2$ if and only if

$$b_0 = -e^{s_0\tau}(s_0^2 + \omega^2), \qquad 2s_0 + \tau(s_0^2 + \omega^2) = 0. \tag{4}$$

The admissibility region is the set $\{(s_0, \tau) \in \mathbb{R} \times (0, +\infty) \mid 2s_0 + \tau(s_0^2 + \omega^2) = 0\}$. For $\omega = 2\pi$, we input $n = 2$ and $m = 0$ in P3δ, select "Control-oriented MID", and choose $a_1 = 0$ and $a_0 = (2\pi)^2 \approx 39.48$. Selecting $s_{0,\min} = -20$ and $\tau_{\max} = 0.2$ for the admissibility plot, we get the admissibility region in Figure 2(a). We then choose $\tau = 0.12$ in P3δ and obtain that $b_0 \approx -33.81$ and $s_0 \approx -2.859$, which agrees with (4). We also obtain the roots of Δ in a given rectangle, represented in Figure 2(b) for the rectangle $\{s \in \mathbb{C} \mid |\mathrm{Re}\, s| \leq 500, |\mathrm{Im}\, s| \leq 500\}$, and time simulations of solutions, for instance the one in Figure 2(c) obtained with the constant initial condition $x(t) = 1$.

4. ONLINE VERSION

Since its creation, P3δ had vocation to be available to the greatest number and on all possible platforms. The current version of the software is available in local executable version, and now the development team wants to propose an online version ready to use in one click. The online version of P3δ is hosted on servers thanks to the *Binder* service (Project Jupyter et al., 2018). *Binder* allows to create instances of personalized computing environment directly from a *GitHub* repository that can be shared and used by users. The *Binder* service is free to use and is powered by *BinderHub*, an open-source tool that deploys the service in the cloud. The online version of P3δ is written in Python and structured as a *Jupyter Notebook*, an open document format which can contain live code, equations, visualizations, and text. The Jupyter Notebook is completed by a friendly user interface built using interactive widgets from Python's `ipywidgets` module.

P3δ online is based on the program and features of the executable version. The first version of the online software includes for the moment features from the "Generic MID", "Control-oriented MID", and "Generic CRRID" modes of P3δ described above in Sections 2.1–2.3, with user inputs similar to those from the executable version.

In the "Generic MID" mode, the online version of P3δ returns the spectrum distribution as well as a normalized quasipolynomial which admits a root of multiplicity $n + m + 1$ at the origin. In the "Control-oriented MID" mode, the online version of P3δ returns the admissibility region, a normalized quasipolynomial which admits a root of multiplicity $m + 2$ at the origin, and an illustration of the bifurcation of the root of multiplicity $m + 2$ with respect to variations of the value of the delay τ. Illustrations of the online version of P3δ are provided in Figure 3.

5. CONCLUSION AND PLANNED DEVELOPMENTS

The current version of P3δ exploits both MID and CRRID properties in its "Generic" mode and only the MID property in its the "Control-oriented" mode. Its main novelties with respect to its previous versions are the design for both retarded as well as neutral equations, the plot of the admissible assignment region in "Control-oriented MID" mode, and an online version of the software. In future developments, inspired from "Control-oriented MID" mode, other configurations for the spectrum distribution guaranteeing the dominancy of an assigned spectral value will be proposed, such as a "Control-oriented CRRID" mode.

ACKNOWLEDGMENTS

This work is partially supported by a public grant overseen by the French National Research Agency (ANR) as part of the "Investissement d'Avenir" program, through the iCODE project funded by the IDEX Paris-Saclay, ANR-11-IDEX0003-02. The authors also acknowledge the support of Institut Polytechnique des Sciences Avancées (IPSA).

The authors wish to acknowledge the work of the full P3δ development team, which, in addition to the authors, also include Mickael Alcaniz, Yoann Audet, Thomas Charbonnet, Honoré Curlier, Ayrton Hammoumou, Julien Huynh, Pierre-Henry Poret, Achrafy Said Mohamed and Franck Sim. The development of P3δ was also made possible thanks to the work of the Cyb'Air Association.

REFERENCES

Amrane, S., Bedouhene, F., Boussaada, I., and Niculescu, S.I. (2018). On qualitative properties of low-degree quasipolynomials: further remarks on the spectral abscissa and rightmost-roots assignment. *Bull. Math. Soc. Sci. Math. Roumanie (N.S.)*, 61(109)(4), 361–381.

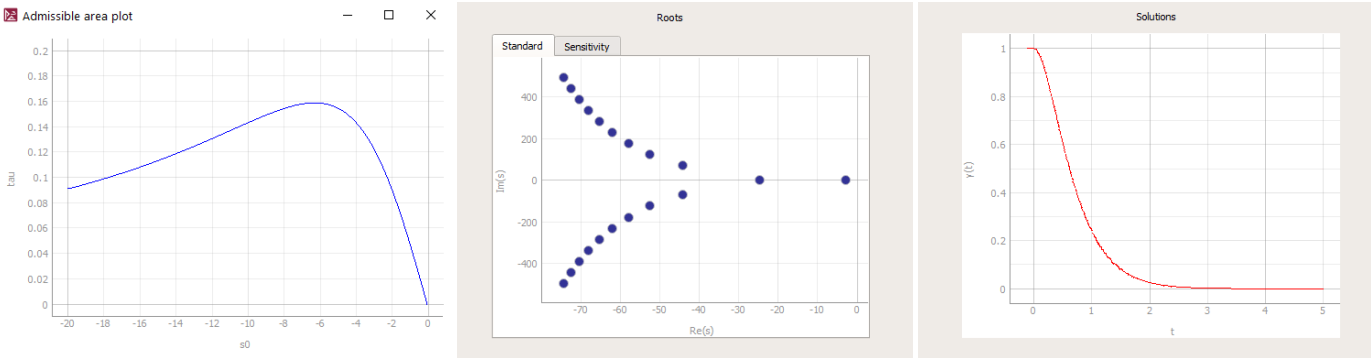
Atay, F.M. (1999). Balancing the inverted pendulum using position feedback. *Appl. Math. Lett.*, 12(5), 51–56.

Avanessoff, D., Fioravanti, A.R., Bonnet, C., and Nguyen, L.H.V. (2014). H_∞-stability analysis of (fractional) delay systems of retarded and neutral type with the Matlab toolbox YALTA. In *Delay systems*, volume 1 of *Adv. Delays Dyn.*, 285–297. Springer, Cham.

Balogh, T., Boussaada, I., Insperger, T., and Niculescu, S.I. (2021). Conditions for stabilizability of time-delay systems with real-rooted plant. Submitted.

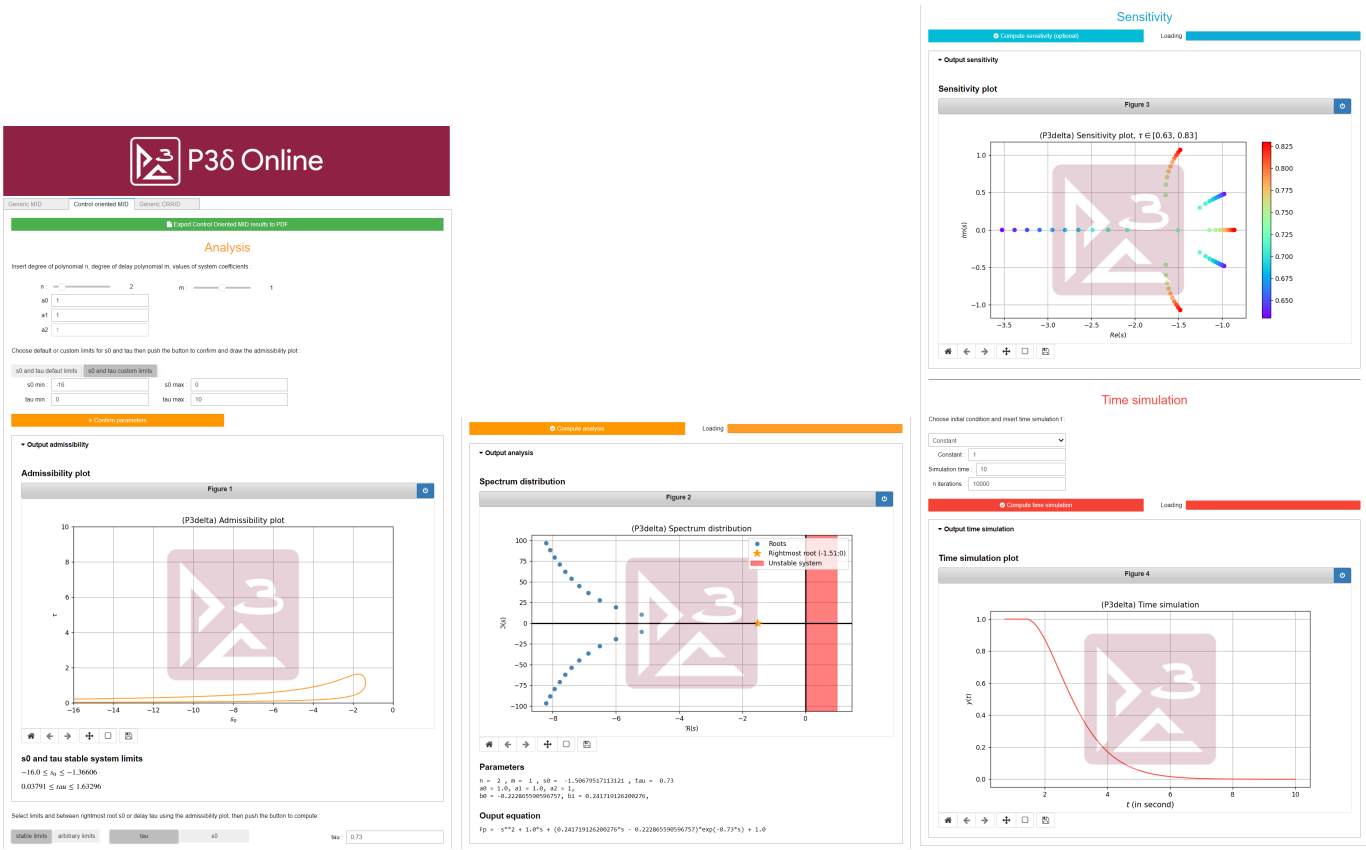
Balogh, T., Insperger, T., Boussaada, I., and Niculescu, S.I. (2020). Towards an MID-based delayed design for arbitrary-order dynamical systems with a mechanical application. *IFAC-PapersOnLine*, 53(2), 4375–4380. 21th IFAC World Congress.

Bedouhene, F., Boussaada, I., and Niculescu, S.I. (2020). Real spectral values coexistence and their effect on the



(a) (b) (c)

Fig. 2. Example of the usage of P3δ in the "Control-oriented MID" mode for the delay-differential equation (3) with $\omega = 2\pi$. (a) Admissible region. (b) Roots of the characteristic quasipolynomial. (b) Numerical simulation of a solution with initial condition $x(t) = 1$ for $t \leq 0$.



(a) (b) (c)

Fig. 3. Example of use of P3δ Online in the "Control-oriented MID" mode. The user can easily select values of n and m using sliders and enter precise values for s_0 and τ. The perturbed spectrum in (c) is computed with respect to variations in the delay value τ. Computations are performed in real time and the user can switch between different figures and results. P3δ Online also generates a PDF report with data and figures.

stability of time-delay systems: Vandermonde matrices and exponential decay. *Comptes Rendus. Mathématique*, 358(9-10), 1011–1032.

Benarab, A., Boussaada, I., Trabelsi, K., Mazanti, G., and Bonnet, C. (2020). The MID property for a second-order neutral time-delay differential equation. In *2020 24th International Conference on System Theory, Control and Computing (ICSTCC)*, 202–207.

Berenstein, C.A. and Gay, R. (1995). *Complex analysis and special topics in harmonic analysis.* Springer-Verlag, New York.

Boussaada, I., Mazanti, G., and Niculescu, S.I. (2021). The generic multiplicity-induced-dominancy property from retarded to neutral delay-differential equations: When delay-systems characteristics meet the zeros of Kummer functions. Submitted.

Boussaada, I., Mazanti, G., Niculescu, S.I., Huynh, J., Sim, F., and Thomas, M. (2020a). Partial pole placement via delay action: A Python software for delayed feedback stabilizing design. In *2020 24th International Conference on System Theory, Control and Computing (ICSTCC)*, 196–201.

Boussaada, I. and Niculescu, S.I. (2018). On the dominancy of multiple spectral values for time-delay systems with applications. *IFAC-PapersOnLine*, 51(14), 55–60. 14th IFAC Workshop on Time Delay Systems.

Boussaada, I., Niculescu, S.I., El-Ati, A., Pérez-Ramos, R., and Trabelsi, K. (2020b). Multiplicity-induced-dominancy in parametric second-order delay differential equations: Analysis and application in control design. *ESAIM Control Optim. Calc. Var.*, 26, Paper No. 57.

Boussaada, I., Tliba, S., Niculescu, S.I., Ünal, H.U., and Vyhlídal, T. (2018). Further remarks on the effect of multiple spectral values on the dynamics of time-delay systems. Application to the control of a mechanical system. *Linear Algebra Appl.*, 542, 589–604.

Breda, D., Maset, S., and Vermiglio, R. (2009). TRACE-DDE: a tool for robust analysis and characteristic equations for delay differential equations. In *Topics in time delay systems*, volume 388 of *Lect. Notes Control Inf. Sci.*, 145–155. Springer, Berlin.

Engelborghs, K., Luzyanina, T., and Roose, D. (2002). Numerical bifurcation analysis of delay differential equations using DDE-BIFTOOL. *ACM Trans. Math. Software*, 28(1), 1–21.

Hale, J.K. and Verduyn Lunel, S.M. (1993). *Introduction to functional differential equations*. Springer-Verlag, New York.

Irofti, D.A., Boussaada, I., and Niculescu, S.I. (2016). On the codimension of the singularity at the origin for networked delay systems. In A. Seuret, L. Hetel, J. Daafouz, and K.H. Johansson (eds.), *Delays and Networked Control Systems*, 3–15. Springer International Publishing.

Ma, D., Boussaada, I., Bonnet, C., Niculescu, S.I., and Chen, J. (2020). Multiplicity-Induced-Dominancy extended to neutral delay equations: Towards a systematic PID tuning based on Rightmost root assignment. In *ACC 2020 - American Control Conference*. Denver, United States.

Manitius, A.Z. and Olbrot, A.W. (1979). Finite spectrum assignment problem for systems with delays. *IEEE Trans. Automat. Control*, 24(4), 541–553.

Mazanti, G., Boussaada, I., and Niculescu, S.I. (2020a). On qualitative properties of single-delay linear retarded differential equations: Characteristic roots of maximal multiplicity are necessarily dominant. *IFAC-PapersOnLine*, 53(2), 4345–4350. 21st IFAC World Congress.

Mazanti, G., Boussaada, I., and Niculescu, S.I. (2021a). Multiplicity-induced-dominancy for delay-differential equations of retarded type. *J. Differential Equations*, 286, 84–118.

Mazanti, G., Boussaada, I., Niculescu, S.I., and Chitour, Y. (2021b). Effects of roots of maximal multiplicity on the stability of some classes of delay differential-algebraic systems: the lossless propagation case. *IFAC-PapersOnLine*, 54(9), 764–769. 24th International Symposium on Mathematical Theory of Networks and Systems (MTNS 2020).

Mazanti, G., Boussaada, I., Niculescu, S.I., and Vyhlídal, T. (2020b). Spectral dominance of complex roots for single-delay linear equations. *IFAC-PapersOnLine*, 53(2), 4357–4362. 21st IFAC World Congress.

Michiels, W., Engelborghs, K., Vansevenant, P., and Roose, D. (2002). Continuous pole placement for delay equations. *Automatica*, 38(5), 747–761.

Michiels, W. and Niculescu, S.I. (2014). *Stability, control, and computation for time-delay systems: An eigenvalue-based approach*. SIAM, Philadelphia, PA, second edition.

Niculescu, S.I., Michiels, W., Gu, K., and Abdallah, C.T. (2010). Delay effects on output feedback control of dynamical systems. In F.M. Atay (ed.), *Complex time-delay systems*, 63–84. Springer, Berlin.

Parini, R. (2018–). cxroots: A Python module to find all the roots of a complex analytic function within a given contour. URL `https://github.com/rparini/cxroots`.

Project Jupyter, Matthias Bussonnier, Jessica Forde, Jeremy Freeman, Brian Granger, Tim Head, Chris Holdgraf, Kyle Kelley, Gladys Nalvarte, Andrew Osheroff, Pacer, M., Yuvi Panda, Fernando Perez, Benjamin Ragan Kelley, and Carol Willing (2018). Binder 2.0 - Reproducible, interactive, sharable environments for science at scale. In Fatih Akici, David Lippa, Dillon Niederhut, and M. Pacer (eds.), *Proceedings of the 17th Python in Science Conference*, 113–120.

Ram, Y., Mottershead, J., and Tehrani, M. (2011). Partial pole placement with time delay in structures using the receptance and the system matrices. *Linear Algebra and its Applications*, 434(7), 1689–1696.

Stépán, G. (1989). *Retarded dynamical systems: stability and characteristic functions*, volume 210 of *Pitman Research Notes in Mathematics Series*. Longman Scientific & Technical, Harlow.

Suh, I.H. and Bien, Z. (1979). Proportional minus delay controller. *IEEE Trans. Automat. Control*, 24(2), 370–372.

Tallman, G.H. and Smith, O.J.M. (1958). Analog study of dead-beat posicast control. *IRE Transactions on Automatic Control*, 4(1), 14–21.

Vyhlídal, T. and Zítek, P. (2014). QPmR—quasi-polynomial root-finder: algorithm update and examples. In *Delay systems*, volume 1 of *Adv. Delays Dyn.*, 299–312. Springer, Cham.

Wielonsky, F. (2001). A Rolle's theorem for real exponential polynomials in the complex domain. *J. Math. Pures Appl. (9)*, 80(4), 389–408.