Reliably computing all characteristic roots of delay differential equations in a given right half plane using a spectral method

Zhen Wu
Wim Michiels

Report TW 596, May 2011



Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

Reliably computing all characteristic roots of delay differential equations in a given right half plane using a spectral method

Zhen Wu
Wim Michiels

Report TW 596, May 2011

Department of Computer Science, K.U.Leuven

Abstract

Spectral discretization methods are well established methods for the computation of characteristic roots of time-delay systems. In this paper a method is presented for computing all characteristic roots in a given right half plane. In particular, a procedure for the automatic selection of the number of discretization points is described. This procedure is grounded in the connection between a spectral discretization and a rational approximation of exponential functions. First, a region that contains all desired characteristic roots is estimated. Second, the number of discretization points is selected in such a way that in this region the rational approximation of the exponential functions is accurate. Finally, the characteristic roots approximations, obtained from solving the discretized eigenvalue problem, are corrected up to the desired precision by a local method. The effectiveness and robustness of the procedure are illustrated with several examples and compared with DDE-BIFTOOL.

Keywords : delay differential equations, characteristic root, spectral discretization, rational approximation.

Reliably computing all characteristic roots of delay differential equations in a given right half plane using a spectral method

Zhen Wu and Wim Michiels
Department of Computer Science, K.U.Leuven
Celestijnenlaan 200A, 3001 Heverlee, Belgium
{Zhen.Wu,Wim.Michiels}@cs.kuleuven.be

May, 2011

Abstract

Spectral discretization methods are well established methods for the computation of characteristic roots of time-delay systems. In this paper a method is presented for computing all characteristic roots in a given right half plane. In particular, a procedure for the automatic selection of the number of discretization points is described. This procedure is grounded in the connection between a spectral discretization and a rational approximation of exponential functions. First, a region that contains all desired characteristic roots is estimated. Second, the number of discretization points is selected in such a way that in this region the rational approximation of the exponential functions is accurate. Finally, the characteristic roots approximations, obtained from solving the discretized eigenvalue problem, are corrected up to the desired precision by a local method. The effectiveness and robustness of the procedure are illustrated with several examples and compared with DDE-BIFTOOL.

Keywords: delay differential equations, characteristic root, spectral discretization, rational approximation

1 Introduction

We consider a linear time-delay system of retarded type [11, 15] described by the delay differential equation (DDE)

$$\dot{x}(t) = A_0 x(t) + \sum_{k=1}^{m} A_k x(t-\tau_k), \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state variable at time t, $A_k \in \mathbb{R}^{n\times n}$, $k = 0, 1, \ldots, m$, are real matrices, and $0 < \tau_1 < \tau_2 < \cdots < \tau_m$ represent the time-delays. For any $\phi \in X$, where $X := \mathcal{C}([-\tau_m, 0], \mathbb{R}^n)$ is the Banach space of continuous functions mapping the interval $[-\tau_m, 0]$ into $\mathbb{R}^n$ and equipped with the supremum norm, a forward solution of (1) is uniquely defined [13]. Denote by $x(\phi) : [-\tau_m, +\infty] \mapsto \mathbb{C}^n$ the forward solution with initial condition ϕ at $t = 0$, i.e., $x(\phi)(\theta) = \phi(\theta),\ \theta \in [-\tau_m,\ 0]$.

The characteristic roots of (1) can be found as the solutions of the nonlinear eigenvalue problem

$$\Delta(\lambda)v = 0,\ \lambda \in \mathbb{C},\ v \in \mathbb{C}^n,\quad v \neq 0, \tag{2}$$

where $\Delta(\lambda)$ is the characteristic matrix,

$$\Delta(\lambda) := \lambda I - A_0 - \sum_{k=1}^{m} A_k e^{-\lambda\tau_k}, \tag{3}$$

see [22, 15, 13]. Several numerical approaches for computing characteristic roots are based on the discretization of either the solution operator associated with (1) (cf. [2, 10]) or the infinitesimal generator of the solution operator (cf. [3, 6, 4, 20]). The solution operator $\mathcal{T}(t)$, $t \geq 0$, associated with (1), is defined by the relation

$$\mathcal{T}(t)\phi = x_t(\phi),$$

where $x_t(\phi)$ is the function segment satisfying $x_t(\phi)(\theta) = x(\phi)(t+\theta)$, $\theta \in [-\tau_m, 0]$. The infinitesimal generator $\mathcal{A}$ of $\mathcal{T}(t)$ has domain

$$\mathcal{D}(\mathcal{A}) = \left\{ \phi \in X : \phi' \in X, \phi'(0) = A_0\phi(0) + \sum_{k=1}^{m} A_k\phi(-\tau_k) \right\}, \tag{4}$$

and its action can be expressed as

$$(\mathcal{A}\phi)(\theta) = \frac{d\phi}{d\theta}(\theta), \quad \theta \in [-\tau_m, 0], \quad \phi \in \mathcal{D}(\mathcal{A}). \tag{5}$$

This allows us to rewrite equation (1) as an abstract ordinary equation,

$$\frac{dx_t}{dt} = \mathcal{A}x_t, \quad x_t \in X. \tag{6}$$

Relations between the characteristic roots and the spectrum of the solution operator $\mathcal{T}$ and its infinitesimal generator $\mathcal{A}$ are described in [7] [13]. The characteristic roots are the solutions of the eigenvalue problem

$$\mathcal{A}\phi = \lambda\phi, \quad \lambda \in \mathbb{C}, \quad \phi \in X, \quad \phi \neq 0. \tag{7}$$

Hence, inferred from the equivalent representation of (1) as (6), the characteristic roots can be computed as either the solutions of the finite-dimensional nonlinear eigenvalue problem (2), or as the solutions of the infinite-dimensional linear eigenvalue problem (7). The characteristic roots and the spectrum of the solution operator $\mathcal{T}(t)$, $\sigma(\mathcal{T}(t))$, are related by

$$\det \Delta(\lambda) = 0 \Leftrightarrow \lambda = \frac{1}{t}\ln\mu, \quad \mu \in \sigma(\mathcal{T}(t))\backslash\{0\}, \ \forall t > 0.$$

The above relations lead to the idea of transferring the problem of computing approximations of characteristic roots to a corresponding eigenvalue problem for a suitable matrix discretization of the solution operator or its infinitesimal generator.

In [10] Engelborghs and Roose proposed a method for computing characteristic roots, based on discretizing the solution operator using a linear multi-step (LMS) time-integration method (see, e.g. [1] for a survey on time-integration of DDEs), and implemented it in the software package DDE-BIFTOOL for the bifurcation and stability analysis (cf. [8] [9]). This package contains a heuristic to determine the step-length in order to compute all characteristic roots in a given right half plane [10]. In [19] Koen Verheyden et al. described an improved step-length heuristic to reduce the size of the algebraic eigenvalue problem. This new heuristic has been incorporated in DDE-BIFTOOL. v.2.03.

In [3] Breda et al. presented an approach for computing characteristic roots, based on a discretization of the infinitesimal generator via a spectral method. The approach was implemented in the software package TRACE-DDE [5]. This software package does not include an automatic choice of the number of discretization points. In [3] it is suggested to determine the number of discretization points based on the bounds describing the local, asymptotic convergence of the individual characteristic roots, however, this requires an accurate knowledge of error constants. In this paper, the relation between a spectral discretization method and a rational approximation is used to derive a procedure that can automatically select the number of discretization points as small as possible, but sufficiently large to compute all characteristic roots in a given right half plane. The approach also exploits the property that the error on individual characteristic roots can be removed by Newton corrections on the nonlinear equations (2). In this way the role of solving the discretized eigenvalue problem reduces to generate sufficiently good estimates for all characteristic roots in the half plane under consideration, instead of fully accurate solutions.

The article is organized as follows. Section 2 is devoted to some preliminary results and the problem formulation. In Sections 3–4 the approach to determine the number of discretization points is described. Numerical results in Section 5 illustrate its effectiveness. Section 6 contains the conclusions. Throughout the paper we make the following assumption.

Assumption 1. *The maximum delay, τ_m, is equal to one.*

Note that Assumption 1 can be made without losing generality: if it is initially not satisfied, one can always resolve this by rescaling the characteristic equation.

Notations

The notations are as follows:

j	: the imaginary identity
λ	: characteristic root
$\mathbb{C}, \mathbb{R}$	: set of the complex and real numbers
$\mathbb{N}$	: set of natural numbers (zero is included)
e	: $e := \exp(1)$
$\det(A)$	: determinant of matrix A
$\sigma(A)$	: spectrum of matrix or operator A
I, I_n	: identity matrix of appropriate dimensions, of dimensions $n \times n$
$\Re(\lambda), \Im(\lambda), \lvert\lambda\rvert,\ \lambda \in \mathbb{C}$	: real part, imaginary part and modulus of λ
$\vec{\omega} \in [0,\ 2\pi)^m, \ldots$	: short notation for $(\omega_1, \ldots, \omega_m)$, ...
$\angle z,\ z \in \mathbb{C}$	: the phase of z, $\angle z \in [-\pi, \pi)$
$\lceil a \rceil, a \in \mathbb{R}$	: integer obtained by rounding a towards infinity
$[a], a \in \mathbb{R}$	: integer obtained by rounding a towards the nearest integer

2 Preliminaries

First, we review the spectral discretization of the infinitesimal generator as in [3], describe its properties, and outline the algorithm used to compute characteristic roots. Subsequently, we describe the motivation and the goal of the improvement pursued in this paper.

2.1 A spectral discretization approach for computing characteristic roots

We consider the spectral discretization of the eigenvalue problem (7), using a mesh Ω_N of $N+1$ distinct points in the interval $[-\tau_m, 0]$,

$$\Omega_N = \{\theta_{N,i},\ i = 1, \ldots, N+1\},$$

with

$$-\tau_m \leq \theta_{N,1} < \ldots < \theta_{N,N+1} = 0.$$

The discretization is based on replacing the space X by the space X_N of discrete functions defined on the grid Ω_N. Specifically, any $\phi \in X$ is discretized into the block vector $\mathbf{x} \in X_N$ with components

$$\mathbf{x}_i = \phi(\theta_{N,i}) \in \mathbb{C}^n, \quad i = 1, \ldots, N+1.$$

Let $\mathcal{L}_N\mathbf{x}$, $\mathbf{x} \in X_N$, be the unique $\mathbb{C}^n$-valued interpolating polynomial of degree smaller or equal to N, satisfying

$$(\mathcal{L}_N\mathbf{x})(\theta_{N,i}) = \mathbf{x}_i, \quad i = 1, \ldots .N+1.$$

Operator $\mathcal{A}$, given by (5), can now be approximated by matrix $A_N : X_N \to X_N$, defined by

$$\begin{aligned}(A_N\mathbf{x})_i &= (\mathcal{L}_N\mathbf{x})'(\theta_{N,i}), \quad i = 1, \ldots, N,\\ (A_N\mathbf{x})_{N+1} &= A_0(\mathcal{L}_N\mathbf{x})(0) + \sum_{k=1}^{m} A_k(\mathcal{L}_N\mathbf{x})(-\tau_k).\end{aligned}$$

By using the Lagrange representation of $\mathcal{L}_N\mathbf{x}$, the explicit expression

$$A_N = \begin{bmatrix} a_{1,1} & \cdots & a_{1,N+1} \\ \vdots & \ddots & \vdots \\ a_{N+1,1} & \cdots & a_{N+1,N+1} \end{bmatrix} \in \mathbb{C}^{n(N+1)\times n(N+1)} \tag{8}$$

is obtained, where

$$a_{il} = \begin{cases} \ell_l'(\theta_{N,i})I_n & l \in \{1, \ldots, N+1\}, \quad i \in \{1, \ldots, N\}, \\ \displaystyle\sum_{k=0}^{m} A_k \ell_l(-\tau_k) & l \in \{1, \ldots, N+1\}, \quad i = N+1. \end{cases} \tag{9}$$

The functions ℓ_l in (9) are the Lagrange polynomials relative to Ω_N, i.e., polynomials of degree N satisfying

$$\ell_l(\theta_{N,i}) = \begin{cases} 1, & i = l, \\ 0, & i \neq l, \end{cases} \qquad i, l \in \{1, \ldots, N+1\}. \tag{10}$$

The discretization allows to approximate the operator eigenvalue problem (7) by the matrix eigenvalue problem

$$A_N \mathbf{x} = \lambda \mathbf{x}, \quad \lambda \in \mathbb{C}, \quad \mathbf{x} \in \mathbb{C}^{n(N+1)}, \quad \mathbf{x} \neq 0. \tag{11}$$

In what follows the grid points in Ω_N are chosen as

$$\theta_{N,i} = \frac{\tau_m}{2}(\alpha_{N,i} - 1), \quad \alpha_{N,i} = -\cos\frac{\pi i}{N+1}, \quad i = 1, \ldots, N+1, \tag{12}$$

for which spectral convergence of eigenvalues of A_N to corresponding eigenvalues of $\mathcal{A}$, i.e., an approximation error $\mathcal{O}(N^{-N})$, is observed. This can be explained by the fact that grid points (12) have the same asymptotic distribution as a grid of Chebyshev extremal points, for which spectral convergence has been proven in [3]. We refer to [17] for the connections between convergence properties and the distribution of grid points. As shown in [14] the use of the grid (12) further allows us to rewrite the eigenvalue problem of A_N as

$$(\Sigma_N - \lambda \Pi_N)\mathbf{c} = 0, \quad \lambda \in \mathbb{C}, \quad \mathbf{c} \in \mathbb{C}^{(N+1)n}, \quad \mathbf{c} \neq 0, \tag{13}$$

where

$$\Pi_N = \frac{\tau_m}{4} \begin{pmatrix} \frac{4}{\tau_m} & \frac{4}{\tau_m} & \frac{4}{\tau_m} & \cdots & & \cdots & \frac{4}{\tau_m} \\ 2 & 0 & -1 & & & & \\ & \frac{1}{2} & 0 & -\frac{1}{2} & & & \\ & & \frac{1}{3} & 0 & \ddots & & \\ & & & \frac{1}{4} & \ddots & -\frac{1}{N-2} & \\ & & & & \ddots & 0 & -\frac{1}{N-1} \\ & & & & & \frac{1}{N} & 0 \end{pmatrix} \otimes I_n \tag{14}$$

and

$$\Sigma_N = \begin{pmatrix} R_0 & R_1 & \cdots & R_N \\ & I_n & & \\ & & \ddots & \\ & & & I_n \end{pmatrix}, \tag{15}$$

with

$$R_i = A_0 + \sum_{k=1}^{m} A_k T_i(-2\frac{\tau_k}{\tau_m} + 1), \quad i = 0, \ldots, N.$$

Here the functions $T_i,\ i = 0, \ldots, N$ are the Chebyshev polynomials of the first kind and order i. Because Π_{N_1} and Σ_{N_1} are sub-matrices of Π_{N_2} and Σ_{N_2} for $N_2 > N_1$, an increase of the number of the discretization points can be dealt with by simply extending the matrices. Note that this property forms the basis of the iterative method described in [14].

Summing up the above results, the characteristic roots of (1) appear either as the solutions of the finite-dimensional nonlinear eigenvalue problem (2) or as the solutions of the infinite-dimensional linear eigenvalue problem (7), which can be discretized into (11) or (13). Both viewpoints can be combined in a computational scheme. The discretization of the linear infinite-dimensional problem allows to obtain estimates for all characteristic roots in a region where the approximation is accurate. These estimates can subsequently be corrected by a local method acting on the nonlinear equations (2). This brings us to the following algorithm.

Algorithm 1. *(Computation of characteristic roots)*

1. *Fix N and compute the eigenvalues of the pencil (Σ_N, Π_N), defined by (14)–(15);*

2. Correct these approximate characteristic roots by applying Newton's method to Equations (2).

We conclude with a central result, on which the analysis in Sections 3–4 strongly relies. The following proposition provides an interpretation of the approximation of $\mathcal{A}$ by A_N, or, equivalently, by the pencil (Σ_N, Π_N), in terms of an approximation of the exponential functions in the characteristic matrix (3).

Proposition 1. *For $\lambda \in \mathbb{C}$, let $p_N(\cdot;\ \lambda)$ be the polynomial of degree N satisfying*

$$\begin{cases} p_N(0;\ \lambda) = 1, \\ p'_N(\theta_{N,i};\ \lambda) = \lambda p_N(\theta_{N,i};\ \lambda), \quad i = 1, \ldots, N. \end{cases} \tag{16}$$

Moreover, let

$$\Delta_N(\lambda) := \lambda I - A_0 - \sum_{k=1}^{m} A_k p_N(-\tau_k;\ \lambda). \tag{17}$$

Then the following statements are equivalent:

$$1. \quad \exists \mathbf{x} \in \mathbb{C}^{(N+1)n},\ \mathbf{x} \neq 0 :\ (\lambda I - A_N)\,\mathbf{x} = 0; \tag{18}$$

$$2. \quad \exists v \in \mathbb{C}^n,\ v \neq 0 :\ \Delta_N(\lambda) v = 0. \tag{19}$$

The proof of Proposition 1 can be found in Section 3 of [3]. In the appendix of [12] it has been shown that the functions

$$\lambda \mapsto p_N(-\tau_i;\ \lambda),\ i = 1, \ldots, m, \tag{20}$$

are proper rational functions of order N, with common poles and with the property of uniform convergence on compact sets to the functions

$$\lambda \mapsto e^{-\lambda \tau_i},\ i = 1, \ldots, m. \tag{21}$$

Note that the latter property is expected since conditions (16) are mixed interpolation / collocation conditions for the function $[-1,\ 0] \ni t \mapsto e^{\lambda t}$. Hence, the effect of the discretization of $\mathcal{A}$ can be interpreted as the effect of a rational approximation of the characteristic matrix.

2.2 Motivation and goal of improvements of previous work

The aim of this work is to derive a procedure for computing *all* characteristic roots in a given right half plane, i.e., the set

$$\{\lambda \in \mathbb{C} :\ \Re(\lambda) \geq r\}, \tag{22}$$

where $r \in \mathbb{R}$. This problem is well posed since the number of characteristic roots of (1) in any right half plane is finite, see Chapter 2 of [15]. When taking a spectral discretization approach combined with local corrections, as described in Section 2.1, the problem can be translated into an automatic selection of the number of discretization points, N, in Algorithm 1. On the one hand, the value of N should not be chosen too large, since (Σ_N, Π_N) are $n(N+1) \times n(N+1)$ matrices, hence, computing all eigenvalues with a general-purpose eigensolver has complexity $\mathcal{O}(nN)^3$. On the other hand, the characteristic roots in the half plane under consideration need to be sufficiently well approximated by eigenvalues of (Σ_N, Π_N), such that Newton's method converges to the desired solutions. The latter imposes a lower bound on the value of N.

Note that determining all characteristic roots in a given half plane is an important problem, e.g., in the context of numerical stability and bifurcation analysis. Moreover, a robust method for the automatic selection of the number of discretization points frees the user of the software from choosing this important parameter. The latter would require specific knowledge about the spectrum of delay equations and about the adopted discretization scheme, and in this way, hinder the dissemination in application areas.

3 Approach to determine the number of discretization points

The proposed procedure is based on the connection between a spectral discretization and a rational approximation of exponential functions, described by Proposition 1. The main steps, discussed in §3.1–§3.2, can be summarized as follows. First a region is determined which contains all characteristic roots. Subsequently, the number of discretization points is selected in such a way that the rational approximation is accurate in this

region. In §3.3 some implementation aspects are addressed. For clarity of the presentation the procedure is explained by means of the problem of computing all characteristic roots in the right half plane

$$\{\lambda \in \mathbb{C} : \ \Re(\lambda) \geq 0\}. \tag{23}$$

The general case (22) can be treated by a preliminary shifting of the origin of the complex plane, i.e., by the substitution $\lambda \leftarrow \lambda - r$.

3.1 Estimate the region containing all characteristic roots in the right half plane

We start with a technical result.

Proposition 2. *All characteristic roots of (1) satisfying $\Re(\lambda) \geq \xi,\ \xi \in \mathbb{R}$, belong to the set*

$$\Omega_\xi := \left\{ \bigcup \sigma \left(A_0 + \sum_{k=1}^{m} A_k z_k \right) : z_k \in \mathbb{C}, |z_k| \leq e^{-\xi \tau_k}, k = 1, \ldots, m \right\} \bigcap \{\lambda \in \mathbb{C} : \ \Re(\lambda) \geq \xi\}. \tag{24}$$

The boundary of Ω_ξ is included in

$$\Psi_\xi := \left(\bigcup_{\vec{\omega} \in [0, 2\pi)^m} \sigma \left(A_0 + \sum_{k=1}^{m} A_k e^{-\xi \tau_k} e^{j\omega_k} \right) \right) \bigcap \{\lambda \in \mathbb{C} : \ \Re(\lambda) \geq \xi\}. \tag{25}$$

Proof. The characteristic equation of (1) can be interpreted as

$$\lambda \in \sigma \left(A_0 + \sum_{k=1}^{m} A_k e^{-\lambda \tau_k} \right). \tag{26}$$

Since $\Re(\lambda) \geq \xi$, we have $|e^{-\lambda \tau_k}| \leq e^{-\xi \tau_k}$ for $k = 1, \ldots, m$. This leads us to the set Ω_ξ. The second assertion follows from the open mapping theorem. A proof can be found in Appendix A.1 of [18]. □

In order to estimate the location of characteristic roots in the right half plane (23) we can directly use the set Ψ_0, i.e., Ψ_ξ for $\xi = 0$. However, this estimate may be very conservative if the real part of λ is large, since its derivation is based on the bounds $|e^{-\lambda \tau_k}| \leq 1, k = 1, \ldots, m$. In our implementation we therefore split up the right half plane in two regions,

$$\{\lambda \in \mathbb{C} : 0 \leq \Re(\lambda) \leq \kappa\}, \quad \{\lambda \in \mathbb{C} : \ \Re(\lambda) \geq \kappa\}, \quad \kappa > 0, \tag{27}$$

which allows us to use the estimate Ψ_0 for the first region and the estimate Ψ_κ for the second region. This brings us to the following corollary.

Corollary 1. *Let $\kappa > 0$. The characteristic roots of (1) in the right half plane (23) belong to a region whose boundary is included in*

$$\Psi := \Psi_{[0,\kappa]} \bigcup \Psi_\kappa, \tag{28}$$

where

$$\Psi_{[0,\kappa]} := \left(\bigcup_{\vec{\omega} \in [0, 2\pi)^m} \sigma \left(A_0 + \sum_{k=1}^{m} A_k e^{j\omega_k} \right) \right) \bigcap \{\lambda \in \mathbb{C} : \ 0 \leq \Re(\lambda) \leq \kappa\} \tag{29}$$

and

$$\Psi_\kappa := \left(\bigcup_{\vec{\omega} \in [0, 2\pi)^m} \sigma \left(A_0 + \sum_{k=1}^{m} A_k e^{-\kappa \tau_k} e^{j\omega_k} \right) \right) \bigcap \{\lambda \in \mathbb{C} : \ \Re(\lambda) \geq \kappa\}. \tag{30}$$

With the following example we illustrate the bounds obtained from Proposition 2 and Corollary 1. The choice of κ in the software will be discussed in §3.3.

Example 1. *Consider the time delay system*

$$\dot{x}(t) = 3.2x(t) - 33.34x(t-1). \tag{31}$$

In Figure 1 the area in the right half plane which is bounded by the black curve is the roots region estimated from Ψ_0, while the area bounded by the imaginary axis and the blue curve is the estimated region from Ψ, for $\kappa = 1.93$. The characteristic roots in the right half plane are indicated by red $(*)$.

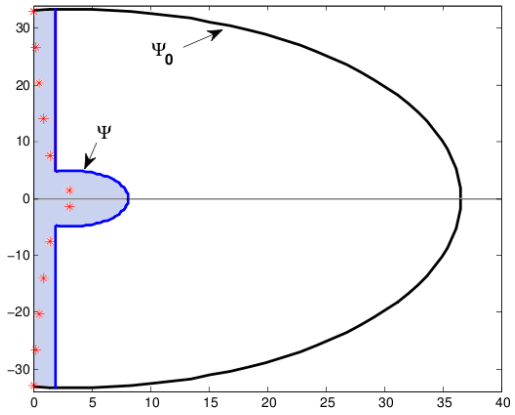


Figure 1: Comparison of Ψ_0 and Ψ.

3.2 Automatic selection of the number of discretization points

Let $\varepsilon > 0$ be a given (small) real number. For any integer $N \in \mathbb{N} \setminus \{0\}$, let the set S_N be defined by

$$S_N := \left\{ \lambda \in \mathbb{C} : \max_{t \in [-1,\ 0]} \frac{|e^{\lambda t} - p_N(t;\ \lambda)|}{|e^{\lambda t}|} < \varepsilon \right\}, \tag{32}$$

where $p_N(\cdot;\ \lambda)$ is the polynomial of degree N, defined by conditions (16). As a first property of the set S_N, the following implication holds,

$$\lambda \in S_N \Rightarrow \frac{\left|e^{-\lambda \tau_k} - p_N(-\tau_k;\ \lambda)\right|}{|e^{-\lambda \tau_k}|} < \varepsilon, \quad k = 1, \ldots, m.$$

Hence, by Proposition 1 the effect of the discretization of $\mathcal{A}$ into A_N can be interpreted as the effect of an approximation of the exponential functions in the characteristic matrix, with a relative accuracy better than ε inside S_N. As a second property, the set S_N is independent of the system matrices and the delays, and can be computed beforehand. Its shape is illustrated in Figure 2 for $\varepsilon = 0.05$ and various values of N.

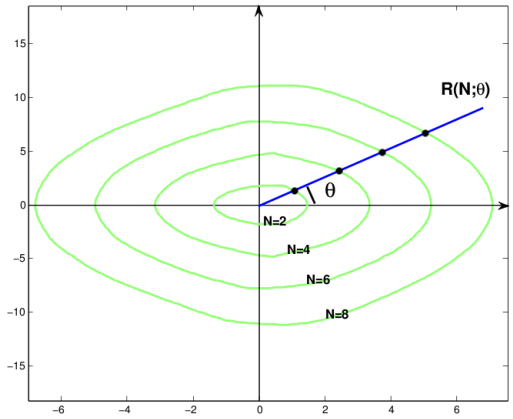


Figure 2: The boundary of the region S_N for different values of N

The automatic selection of the number of discretization points in Algorithm 1 is based on choosing N as small as possible such that the condition

$$\Psi \subseteq S_N \tag{33}$$

is satisfied, where Ψ is given in (28). Figure 3 explains this principle for system (31). In what follows, we give more details about the computation of N.

Because both Ψ and S_N are symmetric with respect to the real axis and we are interested in computing the characteristic roots in the right half plane, we can restrict ourselves to the first quadrant. For $\theta \in [0,\ \pi/2]$

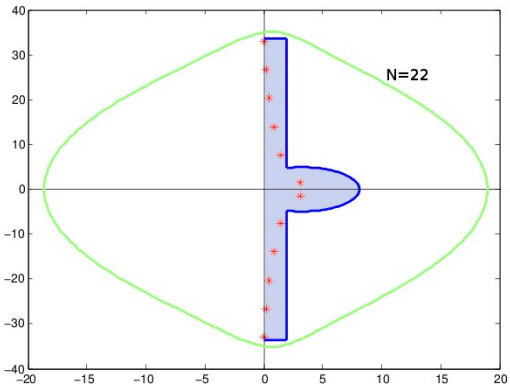


Figure 3: Principle behind the selection of N. The blue shadow area is the region bounded by Ψ. The region bounded by the green curve is S_N, where $N = 22$ is the smallest integer for which (33) holds.

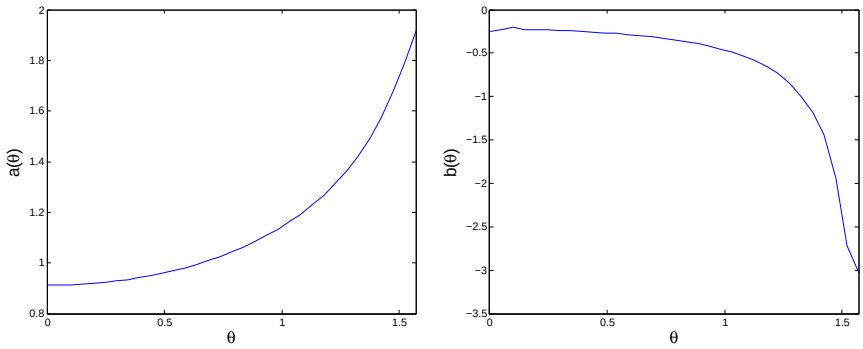
and $N \in \mathbb{N} \setminus \{0\}$ we define

$$R(N;\ \theta) := \left\{ r > 0 :\ re^{j\theta} \in \partial S_N \right\}, \tag{34}$$

where ∂S_N is the boundary of S_N. This quantity is shown in Figure 2. As indicated by this figure, the function $N \mapsto R(N;\ \theta)$ has linear growth in N for all values of θ and, accordingly, (34) can be accurately approximated as

$$R(N;\ \theta) \approx b(\theta) + a(\theta)N. \tag{35}$$

Similarly to the set S_N, the functions a and b do not depend on the system data and can be computed in advance. They are shown in Figure 4.



(a) $a(\theta)$ (b) $b(\theta)$

Figure 4: The functions $\theta \to a(\theta)$ and $\theta \to b(\theta)$, corresponding to $\varepsilon = 0.05$.

For any $z \in \mathbb{C}$ satisfying $\angle z \in [0,\ \pi/2]$, we define N_z as the smallest value of N for which $z \in S_N$. From (35) it follows that

$$N_z = \left\lceil \frac{|z| - b(\angle z)}{a(\angle z)} \right\rceil.$$

Finally, we can compute the minimal value of N such that (33) holds as

$$N = \max\{N_z :\ z \in \Psi,\ \angle z \in [0,\ \pi/2]\}. \tag{36}$$

3.3 Implementation aspects and the choice of parameters

In the definition of S_N, (32), it is not necessary to choose ε very small. The only requirement is that the approximation of the exponential functions by p_N is sufficiently accurate such that the resulting approximation error of the characteristic roots can be removed by Newton's method, in the second step of Algorithm 1. Intensive numerical experiments have led to the value $\varepsilon = 0.05$. Note that for the selected N, ε is an upper

bound on the approximation error of the exponential functions holding for the whole region S_N, while the local error improves exponentially when $|\lambda|$ tends to zero.

In order to evaluate (36), the set Ψ is discretized by restricting $\vec{\omega}$ in (29)–(30) to a grid consisting of p equidistant points in every dimension. This leads to $p^m/2$ grid points on $[0,\ 2\pi)^m$, where the factor $1/2$ stems from the fact that the symmetry of Ψ with respect to the real axis can be exploited. Since for every grid point two eigenvalue problems of size $n \times n$ need to be solved, the computational complexity of determining N is of order

$$\mathcal{O}(p^m n^3). \tag{37}$$

In the software the user can set the value of p. The default value is $p = 20$.

Finally, the choice of κ in the partition (27), on which the set Ψ is based, depends on the choice of p and Ψ_0. It is computed as

$$\kappa = \sin\left(\frac{2\pi}{p}\right) \times \left(\max\{\Re(\lambda) :\ \lambda \in \Psi_0\}\right). \tag{38}$$

Formula (38) is motivated by the following example.

Example 2. *We reconsider system (31). Figure 5 shows the sets Ψ_0 and Ψ, also displayed in Figure 1, along with points obtained by discretizing the interval $[0,\ 2\pi)$ in (29)–(30) where $p = 120$ is chosen for the purpose of visualization. This discretization is used in evaluating (36). On the one hand it is preferable to have κ small, to have an accurate estimate of the roots location and a good choice of N. On the other hand, if κ is taken too small, there is a potential risk that no point of Ψ_0 with real part in the interval $[0,\ \kappa]$ is sampled (i.e., no point of $\Psi_{[0,\kappa]}$), whereas these points are usually critical, as is the case here. The set Ψ_0 is (part of) a circle, where the sampled points lie at an angular distance of $2\pi/p$. Condition (38) then guarantees that at least one sample is present with real part in the interval $[0,\ \kappa]$.*

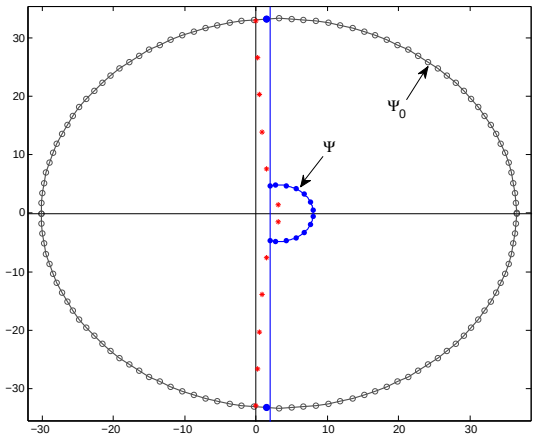


Figure 5: Discretization (sampling) of the sets Ψ_0 and Ψ, for system (31). The blue points are used for the evaluation of (36).

In general the shapes of the sets Ψ_0 and Ψ are much more complex than for the above example, however, it is clear that, to compensate the effects of sampling, a lower bound for κ should behave inversely proportional to p for large p, as is the case for (38). Formula (38) has been validated using a large number of benchmark problems.

4 Time-delay systems with commensurate delays

We show how the interdependence of the delays can be exploited for the case commensurate delays [16], i.e.,

$$\tau_k = n_k \tau, \quad k = 1, \ldots, m, \tag{39}$$

where τ is a positive real number and $n_k \in \mathbb{N},\ k = 1, \ldots, m$. In this way the characteristic matrix becomes

$$\Delta(\lambda) = \lambda I - A_0 - \sum_{k=1}^{m} A_k (e^{-\lambda\tau})^{n_k}. \tag{40}$$

First, we discuss the adaption of the approach to determine N in commensurate delays case. Second, we show how treating delays by commensurate delays may reduce the computational cost of locating characteristic roots for particular cases.

4.1 Adaptation of the method

The following results can be proven similarly to Proposition 2 and Corollary 1.

Proposition 3. *Assume that the delays satisfy (39). All characteristic roots satisfying $\Re(\lambda) \geq \xi$, $\xi \in \mathbb{R}$, belong to the set*

$$\Omega_{c_\xi} := \left\{ \bigcup \sigma \left(A_0 + \sum_{k=1}^{m} A_k z^{n_k} \right) : z \in \mathbb{C}, |z| \leq e^{-\xi\tau} \right\} \bigcap \{\lambda \in \mathbb{C} : \ \Re(\lambda) \geq \xi\}. \tag{41}$$

The boundary of Ω_{c_ξ} is included in

$$\Psi_{c_\xi} = \left(\bigcup_{\omega \in [0,2\pi)} \sigma \left(A_0 + \sum_{k=1}^{m} A_k e^{-\xi\tau n_k} e^{j\omega n_k} \right) \right) \bigcap \{\lambda | \Re(\lambda) \geq \xi\}. \tag{42}$$

Corollary 2. *Assume that the delays satisfy (39). Let $\kappa > 0$. All characteristic roots in the right half plane (23) belong to a region whose boundary is included in*

$$\Psi_c = \Psi_{c_{[0,\kappa]}} \bigcup \Psi_{c_\kappa}, \tag{43}$$

where $\Psi_{c[0,\kappa]}$ and Ψ_{c_κ} are given by

$$\Psi_{c_{[0,\kappa]}} = \left(\bigcup_{\omega \in [0,2\pi)} \sigma \left(A_0 + \sum_{k=1}^{m} A_k e^{j\omega n_k} \right) \right) \bigcap \{\lambda \in \mathbb{C} : \ 0 \leq \Re(\lambda) \leq \kappa\} \tag{44}$$

and

$$\Psi_{c_\kappa} = \left(\bigcup_{\omega \in [0,2\pi)} \sigma \left(A_0 + \sum_{k=1}^{m} A_k e^{-\kappa\tau n_k} e^{j\omega n_k} \right) \right) \bigcap \{\lambda \in \mathbb{C} : \ \Re(\lambda) \geq \kappa\}. \tag{45}$$

Using the above results, the procedure for determining N, discussed in §3.2 can be adapted, where the difference consists of replacing the estimate Ψ by Ψ_c with κ determined as

$$\kappa = \sin\left(\frac{2\pi}{p}\right) \times (\max\{\Re(\lambda) : \ \lambda \in \Psi_{c_0}\}).$$

The consequences are two-fold.

1. In the discretization of (44) and (45) only a grid on the interval $[0, 2\pi)$ is needed. However, working with p grid points per period of the function to be approximated, as in §3.3, brings us to pn_m (where n_m is from $\tau_m = n_m\tau$) grid points over the interval $[0, 2\pi)$ because the function $\omega \mapsto e^{j\omega n_m}$ has period $2\pi/n_m$. Accordingly, the computational complexity of determining N changes from (37) to
$$\mathcal{O}(pn_m n^3). \tag{46}$$
2. The estimated region of the characteristic roots obtained from Ψ_c is in general more accurate than the estimate obtained from Ψ, because the interdependence of the delays is exploited. For comparison, note that $e^{j\omega_1}, \ldots, e^{j\omega_m}$ in (29)-(30) are treated as independent variables.

The latter property is illustrated with the following example.

Example 3. *Consider the equation*

$$\dot{x}(t) = -4x(t) - 5x(t - 0.5) - 5x(t - 1).$$

We take $p = 20$, consequently, $\kappa = 0.309\,(\max\{\Re(\lambda) : \lambda \in \Psi_0 \text{ or } \Psi_{c_0}\})$

- *Treating the delays as commensurate delays, $\tau_{1,2} = 0.5n_{1,2}$, for $n_{1,2} = 1, 2$, yields:*

$$\Psi_c = \Psi_{c_{[0,\kappa]}} \bigcup \Psi_{c_\kappa},$$

where $\Psi_{c[0,\kappa]}$ and Ψ_{c_κ} are given by

$$\Psi_{c_{[0,\kappa]}} = \left(\bigcup_{\omega \in [0,2\pi)} \sigma\left(-4 - 5e^{j\omega} - 5\left(e^{j\omega}\right)^2\right) \right) \bigcap \{\lambda \in \mathbb{C} : \ 0 \leq \Re(\lambda) \leq \kappa\}$$

and

$$\Psi_{c_\kappa} = \left(\bigcup_{\omega \in [0,2\pi)} \sigma\left(-4 - 5e^{-0.5\kappa}e^{j\omega} - 5e^{-\kappa}e^{2j\omega}\right) \right) \bigcap \{\lambda \in \mathbb{C} : \ \Re(\lambda) \geq \kappa\}.$$

- *Treating $\tau_1 = 0.5$, $\tau_2 = 1$ as independent delays, yields:*

$$\Psi := \Psi_{[0,\kappa]} \bigcup \Psi_\kappa$$

where

$$\Psi_{[0,\kappa]} := \left(\bigcup_{(\omega_1,\omega_2) \in [0,2\pi)^2} \sigma\left(-4 - 5e^{j\omega_1} - 5e^{j\omega_2}\right) \right) \bigcap \{\lambda \in \mathbb{C} : \ 0 \leq \Re(\lambda) \leq \kappa\}$$

and

$$\Psi_\kappa := \left(\bigcup_{(\omega_1,\omega_2) \in [0,2\pi)^2} \sigma\left(-4 - 5e^{-0.5\kappa}e^{j\omega_1} - 5e^{-\kappa}e^{j\omega_2}\right) \right) \bigcap \{\lambda \in \mathbb{C} : \ \Re(\lambda) \geq \kappa\}.$$

In Figure 6 both the sets Ψ and Ψ_c are displayed.

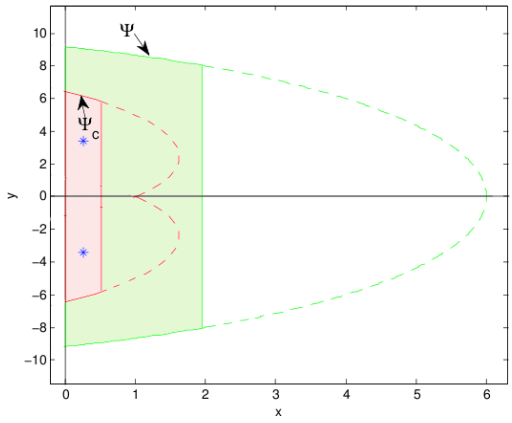


Figure 6: The set Ψ bounds the green shadowed area, the set Ψ_c bounds the red shadowed area. The dashed curves corresponds to Ψ_0 (green curve) and Ψ_{c_0} (red curve).

4.2 Approximation by commensurate delays for large values of m or n_m

The computational complexity of determining N for independent delays and commensurate delays is discussed in (37) and (46) respectively. Hence, the method of determining N proposed in previous sections is particularly suitable if either m or n_m is small. If m or n_m is large, we can approximate the delays as multiples of a given number in the estimation of the roots' location: Given $q \in \mathbb{N} \setminus \{0\}$, we set a new basic delay $\tau_{new} = \tau_m/q$, and (re)calculate the associated integer $n_{k_{new}}$, i.e.:

$$n_{k_{new}} = [\tau_k/\tau_{new}], \ \text{ for } \ k = 1, \ldots, m.$$

Hence, the original delays are approximated as

$$\tau_{k_{new}} = n_{k_{new}} \tau_{new}, \text{ for } k = 1, \ldots, m.$$

In the software this modification is done for independent delays if $m > 3$ and for commensurate delays if $n_m > 100$. The default value for q is $q = 100$. Note that, taking into account Assumption 1, we always get $\tau_{new} = 0.01$ and $n_{m_{new}} = 100$ in such a situation.

It is important to mention that in discretizing $\mathcal{A}$, in the computation of its eigenvalues and in the Newton corrections the exact delay values are used, whereas the delay approximations are only used in estimating the location of characteristic roots.

Example 4. *Consider a system with the following delays:* $\tau_1 = 0.5743$, $\tau_2 = 0.6753$, $\tau_3 = 0.8752$, $\tau_4 = 0.9390$, $\tau_5 = 0.9815$, $\tau_6 = 1.0000$.

- If we consider them as independent delays, the cost of selecting N has complexity $\mathcal{O}\left(p^6 n^3\right)$, according to (37).
- If we consider them as commensurate delays with basic delay $\tau = 0.0001$, consequently $n_6 = 10000$, the computational cost is $\mathcal{O}\left(10^4 p n^3\right)$, according to (46).
- If we set $\tau_{new} = 0.01$ and $n_{m_{new}} = 100$ to approximate the original delays, i.e., approximating them by $\tau_{1_{new}} = 0.57$, $\tau_{2_{new}} = 0.68$, $\tau_{3_{new}} = 0.88$, $\tau_{4_{new}} = 0.94$, $\tau_{5_{new}} = 0.98$, $\tau_{6_{new}} = 1$, the computational cost of selecting N is reduced to $\mathcal{O}\left(10^2 p n^3\right)$.

5 Numerical examples

We present examples to illustrate the effectiveness of the proposed procedure for computing all the characteristic roots in a given right half plane.

Example 5. ***A system of DDE with one delay*** *which has been used in [19]. In this system,* $\tau_1 = 1$,

$$A_0 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -10 & -4 \\ 0 & 0 & 4 & -10 \end{bmatrix} \quad \textit{and} \quad A_1 = \begin{bmatrix} 3 & 3 & 3 & 3 \\ 0 & -1.5 & 0 & 0 \\ 0 & 0 & 3 & -5 \\ 0 & 5 & 5 & 5 \end{bmatrix}.$$

We aim at computing all characteristic roots of Example 5 with $\Re(\lambda) \geq -1.5$. Our procedure automatically chooses the number of discretization points as $N = 19$. Figure 7 shows that this number of N is sufficient. Also note that the selected N is very accurate: N is sufficient to compute all desired roots (i.e., with $\Re(\lambda) \geq -1.5$), however, not large enough to compute all roots with real part satisfying $\Re(\lambda) \geq -1.7$.

We compare the proposed method with the stability routine for equilibria of DDE-BIFTOOL v.2.00 and DDE-BIFTOOL v.2.03, in terms of the size of the discretized eigenvalue problem to be solved. Table 1 shows that the proposed procedure outperforms DDE-BIFTOOL 2.00 and leads to slightly better results than DDE-BIFTOOL 2.03.

Table 1: Size of eigenvalue problem for $\Re(\lambda) \geq r$

r	DDE-BIFTOOL.v.2.00	DDE-BIFTOOL.v.2.03	spectral method
0	184	48	16
-0.5	240	48	28
-1	332	60	36
-1.5	476	96	80
-2	716	140	136
-2.5	1104	220	204
-3	1740	360	340

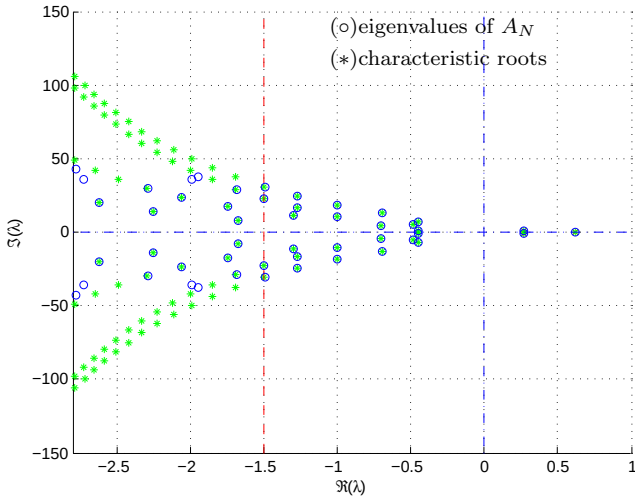


Figure 7: Characteristic roots with $\Re(\lambda) \geq -1.5$.

Example 6. ***A system of DDE with three delays*** *where* $\tau_1 = 0.1$, $\tau_2 = 0.15$, $\tau_3 = 0.25$,

$$A_0 = \begin{bmatrix} -9.6713 & -9.7546 & -9.4913 \\ 1.8381 & 1.7961 & 9.5716 \\ 1.3647 & -2.7957 & -7.3561 \end{bmatrix}, \quad A_1 = \begin{bmatrix} 1.0115 & -9.3006 & 5.3222 \\ 7.2688 & -1.1960 & 9.9968 \\ 3.6508 & -1.2035 & -4.8507 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 7.7163 & 4.5911 & -5.5072 \\ -9.0056 & -0.0260 & -7.5404 \\ -3.3669 & 0.9332 & -0.2958 \end{bmatrix}, \quad A_3 = \begin{bmatrix} 7.4808 & -7.2571 & 9.4377 \\ 2.8285 & -7.1768 & -1.4221 \\ -1.0353 & 9.6519 & 5.1208 \end{bmatrix}.$$

To compute all characteristic roots of Example 6 with $\Re(\lambda) \geq -7$, the proposed procedure selects $N = 15$. Figure 8 shows that this value of N is sufficient, however, it is also sufficient to compute all roots with real part larger than -10. The reason is that, when the number of delays increases, the estimation of the characteristic roots' location becomes more conservative.

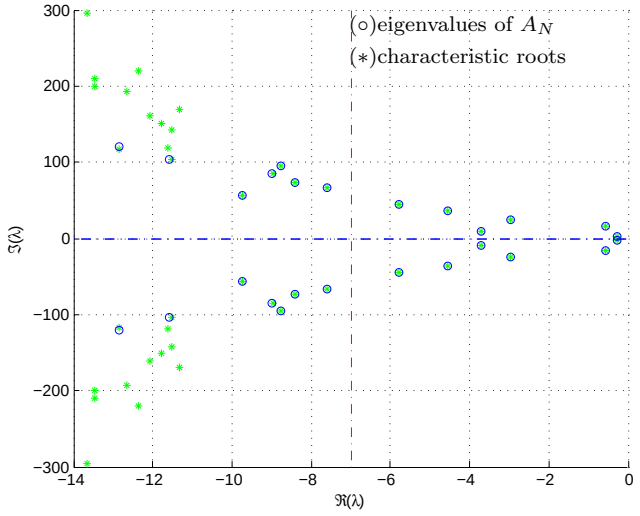


Figure 8: Characteristic roots with $\Re(\lambda) \geq -7$

Again we compared the proposed method with DDE-BIFTOOL v.2.00 and DDE-BIFTOOL v.2.03. In addition, we have also exploited the property that the delays in this example are commensurate delays:

$$\tau_{1,2,3} = 0.05 n_{1,2,3}, \text{ where } n_{1,2,3} = 2, 3, 5.$$

Table 2: Size of eigenvalue problem for $\Re(\lambda) \geq r$

r	DDE-BIFTOOL v.2.00	DDE-BIFTOOL v.2.03	spectral method, independent delays	spectral method, commensurate delays
-2	138	36	24	21
-3	156	36	27	24
-4	180	36	30	27
-5	207	36	33	30
-6	240	42	39	33
-7	282	48	48	39
-8	336	54	57	51
-9	399	66	66	57
-10	480	78	81	72

Table 2 shows the sizes of discretized eigenvalue problems to be solved.

Example 7. ***A system of DDE with nine delays*** *and coefficient matrices of size* 20×20*, i.e.* $m = 9$*,* $n = 20$*. The data of coefficient matrices can be downloaded at*

`http://twr.cs.kuleuven.be/research/software/delay-control/roots/`

The delays are

$$\tau_1 = 0.0087,\ \tau_2 = 0.3000,\ \tau_3 = 0.8625,\ \tau_4 = 0.9848,\ \tau_5 = 1.1580,$$

$$\tau_6 = 1.5009,\ \tau_7 = 1.6103,\ \tau_8 = 1.6830,\ \tau_9 = 1.7147.$$

We use the proposed approach and DDE-BIFTOOL. v.2.03 to compute all characteristic roots with $\Re(\lambda) \geq -1$. As shown in Figure 9, our proposed procedure captures all the desired characteristic roots in

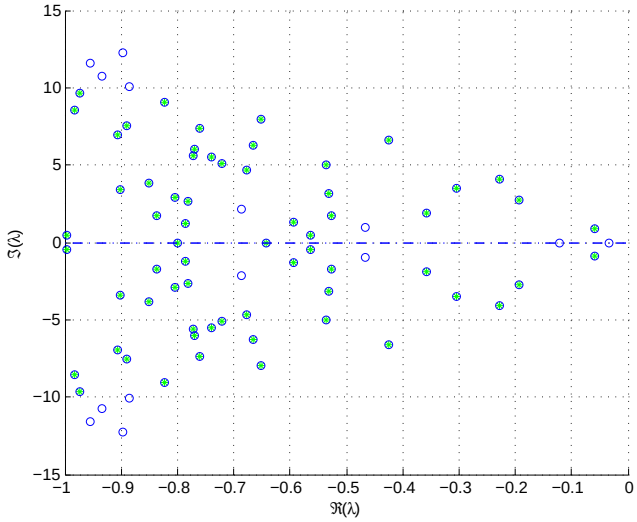


Figure 9: Characteristic roots with $\Re(\lambda) \geq -1$: characteristic roots ($\circ$ in blue) are computed by a spectral method with automatic selection of N; characteristic roots ($*$ in green) are computed by DDE-BIFTOOL.v.2.03

the given right half plane, while DDE-BIFTOOL.v.2.03 misses several desired roots and even the rightmost root. The reason lies in the large number of characteristic roots. The discretization of $\mathcal{T}(h)$ (where $h = 1/N$, N denotes the number of grid points in $[-\tau_m,\ 0]$) in DDE-BIFTOOL leads to approximations of $e^{\lambda h}$, where λ is the characteristic root. Hence, for small h the eigenvalues of $\mathcal{T}(h)$ are clustered around one, making the computation by discretizing $\mathcal{T}(h)$ numerically unstable.

6 Concluding remarks

A spectral discretization of the infinitesimal generator of the solution operator is an established method for computing characteristic roots of linear time-delay systems. We proposed a procedure which can automatically select the number of discretization points, N, such that by solving the discretized eigenvalue problem, followed by Newton corrections on the resulting approximations, all roots in a given right half plane are returned. This makes the method very suitable in the context of stability assessment and bifurcation analysis, and it facilitates the use of the software.

Because from a computational point of view it is important not to overestimate the required number of discretization points, we have preferred developing and validating an accurate but reliable heuristic over constructing a (conservative) lower bound with a hard guarantee that all roots are found. However, the selection procedure has a solid mathematical foundation and builds on the relation between a spectral discretization and a rational approximation, where the heuristic aspect is reduced to the choice of one parameter, ε.

The step-length selection turns out to be very reliable. The code has been tested extensively and in all cases all characteristic roots in the half plane under consideration were found. In terms of the size of the discretized eigenvalue problem to be solved, the method slightly improves the stability routine of DDE-BIFTOOL 2.03, which relies on discretizing the solution operator. The cost of determining the number of discretization points, which is dominated by locating the characteristic roots, is comparable if the delays are treated as independent variables (which is always the case in DDE-BIFTOOL). We have shown how improvements can be made when the delays are commensurate. Formulae (37) and (46) make it clear that the automatic selection of N is particularly suitable for systems with a small number of delays or with commensurate delays where the ratio between the largest delay and the basic delay (the value n_m in (39)), is small. However, problems with large m or n_m can be addressed by approximating the delays by commensurate delays in estimating the roots location, as we have seen.

The advantages of a discretization of the infinitesimal generator of the solution operator over a discretization of the solution operator, as in DDE-BIFTOOL, are two-fold. First, the proposed method yields directly characteristic roots approximations, while DDE-BIFTOOL is based on discretizing $\mathcal{T}(h)$, where $h = 1/N$, with N also denoting the number of grid points used in the interval $[-\tau_m,\ 0]$. For small h this leads to a clustering of the eigenvalues of $\mathcal{T}(h)$ around one, which may lead to a decrease of accuracy as we have illustrated. Second, the first order form (6) is more suitable in the context of control problems (e.g., stabilization, $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control), as control theory largely builds on a standard state space representation of the system.

Finally, the principle used in §3.2 extends to compute all characteristic roots in any given compact set. The software corresponding to this article includes a routine for computing all characteristic roots in a rectangular region. Note that by subdivision, all roots in a very large region can be obtained, see, e.g. the problems addressed in [21].

An implementation of the proposed method is available at

`http://twr.cs.kuleuven.be/research/software/delay-control/roots/`

Acknowledgements

This work has been supported by the Programme of Interuniversity Attraction Poles of the Belgian Federal Science Policy Office (IAP P6- DYSCO), by OPTEC, the Optimization in Engineering Center of the K.U.Leuven, by the project STRT1- 09/33 of the K.U.Leuven Research Council and the project G.0712.11N of the Research Foudation - Flanders (FWO).

References

[1] A. Bellen and M. Zennaro. *Numerical methods for delay differential equations*. Oxford University Press, 2003.

[2] D. Breda. Solution operator approximation for delay differential equation characteristic roots computation via Runge-Kutta methods. *Applied Numerical Mathematics*, 56:305–317, 2006.

[3] D. Breda, S. Maset, and R. Vermglio. Pseudospectral differencing methods for characteristic roots of delay differential equations. *SIAM Journal on Scientific Computing*, 27(2):482–495, 2005.

[4] D. Breda, S. Maset, and R. Vermiglio. Computing the characteristic roots for delay differential equations. *IMA Journal of Numerical Analysis*, 24:1–19, 2004.

[5] D. Breda, S. Maset, and R. Vermiglio. TRACE-DDE: A tool for robust analysis and characteristic equations for delay differential equations. In J.J. Liseau et al., editor, *Topics in Time Delay Systems: Analysis, Algorithm and Control*, pages 145–155. Springer, 2009.

[6] E.A. Butcher and O.A. Bobrenkov. On the chebyshev spectral continuous time approximation for constant and periodic delay differential equations. *Communications in Nonlinear Science and Numerical Simulation*, 16:1541–1554, 2011.

[7] O. Diekmann, S.A. Van Gils, S.M. Verduyn Lunel, and H.O. Walther. *Delay Equations. Functional, Complex and Nonlinear Analysis*. Spring-Verlag, 1995.

[8] K. Engelborghs, T. Luzyanina, and D. Roose. Numerical bifurcation analysis of delay differential equations using DDE-BIFTOOL. *ACM transactions on Mathematical Software*, 28:1–21, 2002.

[9] K. Engelborghs, T. Luzyanina, and G. Samaey. DDE-BIFTOOL v. 2.00: a MATLAB package for bifurcation analysis of delay differential equations. TW Report 330, Department of Computer Science, Katholieke Universiteit Leuven, Belgium, October 2001.

[10] K. Engelborghs and D. Roose. On stability of LMS methods and characteristic roots of delay differential equations. *SIAM Journal on Numerical Analysis*, 40:629–650, 2002.

[11] K. Gu, V.L. Kharitonov, and J. Chen. *Stability of time-delay systems*. Birkhauser, 2003.

[12] S. Gumussoy and W. Michiels. A predictor-corrector type algorithm for the pseudospectral abscissa computation of time-delay systems. *Automatica*, 46(4):657–664, 2010.

[13] J. Hale and S.M. Verduyn Lunel. *Introduction to functional differential equations*. Springer-Verlag, 1993.

[14] E. Jarlebring, K. Meerbergen, and W. Michiels. An arnoldi like method for the delay eigenvalue problem. TW Report 558, Department of Computer Science, Katholieke Universiteit Leuven, Belgium, January 2010.

[15] W. Michiels and S.-I. Niculescu. *Stability and stabilization of time-delay systems. An eigenvalue based approach*. SIAM, 2007.

[16] S.-I. Niculescu. *Delay effects on stability. A robust control approach*, volume 269 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, 2001.

[17] L.N. Trefethen. *Spectral methods in MATLAB*, volume 10 of *Software, Environments, and Tools*. SIAM, 2000.

[18] K. Verheyden. *Numerical bifurcation analysis of large-scale delay differential equations*. PhD thesis, Department of Computer Science, K.U.Leuven, 2007.

[19] K. Verheyden, T. Luzyanina, and D. Roose. Efficient computation of characteristic roots of delay differential equations using LMS methods. *Journal of Computational and Applied Mathematics*, 214:209–226, 2008.

[20] K. Verheyden and D. Roose. Efficient numerical stability analysis of delay equations: a spectral method. In *Proceedings of the IFAC Workshop on Time-Delay Systems*, pages 209–214, 2004.

[21] T. Vyhlídal and P. Zítek. Mapping based algorithm for large-scale computation of quasi-polynomial zeros. *IEEE Transactions on Automatic Control*, 54(1):171–177, 2009.

[22] Vyhlídal, T. *Analysis and synthesis of time delay system spectrum*. PhD thesis, Department of Mechanical Engineering, Chech Technical University at Prague, 2003.