METHODS FOR NUMERICAL COMPUTATION OF CHARACTERISTIC ROOTS FOR DELAY DIFFERENTIAL EQUATIONS: EXPERIMENTAL COMPARISON.

D. Breda

Received March 26, 2003

Abstract. This paper is a collection of tests about the numerical computation of characteristic roots for linear delay differential equations (DDEs) with multiple discrete and distributed delays. Two different approaches are tested, based on the discretization of the infinitesimal generator of the solution operators semigroup associated to the DDE and of the solution operator itself. These approaches are implemented using different numerical techniques such as Runge-Kutta (RK), linear multistep (LMS) and spectral methods.

1 Introduction Let us consider an m-dimensional linear (or linearized) DDE with multiple discrete and distributed delays:

$$y'(t) = L_0 y(t) + \sum_{l=1}^{k} L_l y(t-\tau_l) + \int_{-t_1}^{-t_2} M(\theta) y(t+\theta) d\theta, \quad t \geq 0 \tag{1}$$

where $L_0, L_1, \ldots, L_k \in \mathbf{C}^{m\times m}$, $\tau = \tau_k > \ldots > \tau_1 > \tau_0 = 0$, $t_1 > t_2 \geq 0$ and $M : [-t_1, -t_2] \to \mathbf{C}^{m\times m}$ is a sufficiently smooth function. Without loss of generality we consider the case where $t_1 = \tau_{l_1}$ and $t_2 = \tau_{l_2}$ for some l_1 and l_2 varying from 0 to k.

The characteristic equation associated with (1) is

$$\det(\Delta(\lambda)) = 0 \tag{2}$$

where

$$\Delta(\lambda) := \lambda I - L_0 - \sum_{l=1}^{k} L_l e^{-\lambda \tau_l} - \int_{-t_1}^{-t_2} M(\theta) e^{\lambda\theta} d\theta, \ \lambda \in \mathbf{C} \tag{3}$$

The asymptotic stability of the zero solution of (1) is determined by the rightmost root of (2). In particular the zero solution is asymptotically stable iff this root has negative real part.

The obvious choice to apply a root-finder for non-linear equations to (2) is not a suitable one since the roots are very sensitive to perturbations in the coefficients of the characteristic equation.

Engelborghs, Roose *et al.* proposed in [6], [7] and [8] a method to compute the rightmost characteristic roots based on a LMS time integration of (1) in the case $M = 0$. This solution operator approach avoids the use of the characteristic equation and compute approximations of the roots from a large, standard and sparse eigenvalue problem via a logarithmic transformation.

2000 *Mathematics Subject Classification.* 65L15, 65Q05.
Key words and phrases. Delay differential equations, characteristic roots, large eigenvalues problems.

Breda, Maset and Vermiglio proposed in [2] and [3] other methods (in the case $M \neq 0$ too) which avoids the use of the characteristic equation and they are based on discretizing the infinitesimal generator associated to the semigroup of solution operators for (1), by using RK and LMS (backward differentiation formulae, BDF) methods, respectively. As in [8] the approximations of the roots are eigenvalues of a large sparse matrix, but any transformation is needed.

Moreover the infinitesimal generator approach is presented in [1] by using spectral techniques and the solution operator approach implemented with RK and BDF methods in the single delay case is proposed in [4].

This work collects a number of numerical tests on several DDEs which permits a comparison of the performances and an experimental analysis of some computational aspects of all the above algorithms implemented via *MATLAB* codes. To this aim, the paper is not intended to give neither a detailed description (but only a brief-comprehensive one) nor proofs of convergence of all the methods. For this we refer the interested reader to the relative references.

The paper is organized as follows. Section 2 gives an introduction about the semigroup of solution operators and its infinitesimal generator for (1). Section 3 and 4 briefly describe the numerical methods in the single delay case and their extensions to the multiple and distributed cases, respectively. Numerical results illustrate the comparison in section 5 and we conclude in section 6.

2 Solution operator and infinitesimal generator approaches Let $X := C([-\tau, 0], \mathbf{C}^m)$ equipped with the maximum norm

$$\|\varphi\| = \max_{\theta \in [-\tau,0]} |\varphi(\theta)|, \quad \varphi \in X.$$

It is well-known that the family $\{T(t)\}_{t \geq 0}$ of linear bounded operators on the Banach space X defined by

$$T(t)\varphi := y_t, \quad \varphi \in X \tag{4}$$

is a C_0-semigroup (see Diekmann, van Gils, Verduyn Lunel and Walther [5]). Here y_t is the function

$$y_t(\theta) = y(t+\theta), \quad \theta \in [-\tau, 0]$$

where y is the solution of (1) with initial data

$$y(t) = \varphi(t), \quad -\tau \leq t \leq 0. \tag{5}$$

The infinitesimal generator $\mathcal{A} : D(\mathcal{A}) \to X$ of the semigroup is the unbounded closed operator given by:

$$\begin{aligned} D(\mathcal{A}) = \{\varphi \in X \mid \varphi' \in X \text{ and } \varphi'(0) &= L_0\varphi(0) + \sum_{l=1}^{k} L_l \varphi(-\tau_l) + \\ &+ \int_{-t_1}^{-t_2} M(\theta)\varphi(\theta)d\theta\} \end{aligned} \tag{6}$$

$$\mathcal{A}\varphi = \varphi', \ \varphi \in D(\mathcal{A}) \tag{7}$$

Since the characteristic roots of (1) constitute the spectrum $\sigma(\mathcal{A})$ of $\mathcal{A}$, the infinitesimal generator approach consists in approximating $\mathcal{A}$ by a matrix and then compute its eigenvalues. This is the approach proposed in [1], [2] and [3] using spectral, RK and BDF methods, respectively.

On the other hand, the spectrum $\sigma(T(t))$ of the solution operator $T(t)$ is given by

$$\sigma(T(t)) \setminus \{0\} = e^{t\sigma(\mathcal{A})} \tag{8}$$

and so the characteristic roots can be approximated by computing the eigenvalues of a numerical approximation of $T(t)$ and then taking the logarithm. This is the solution operator approach proposed in [4] and [8] using RK and BDF methods, respectively.

3 Single delay case Let us consider the system of DDEs (1) in the case of one fixed, discrete delay τ:

$$y'(t) = L_0 y(t) + L_1 y(t-\tau). \tag{9}$$

3.1 The solution operator approach: BDF methods For fixed N, N positive integer, let us consider the constant stepsize mesh Ω_N on the interval $[-\tau, 0]$:

$$\begin{cases} \Omega_N = \{\theta_i = -ih \mid i = 0, \dots, N-1\} \\ h = \frac{\tau}{N} \end{cases}$$

and replace the continuous space X by the discrete space $X_N = (\mathbf{C}^m)^{\Omega_N} = \mathbf{C}^{mN}$.

Applying to (9) a general k-steps ($k \le N$) BDF method we obtain an approximation of $y_{t_n}(0)$ at time $t_n = nh$ for $n \in \mathbf{N}$ given by

$$y_n = -\sum_{i=0}^{k-1} \alpha_i B(hL_0) y_{n+i-k} + h\beta_k B(hL_0) L_1 y_{n-N}$$

where

$$B(z) = (\alpha_k - \beta_k z)^{-1}$$

is the stability function of the BDF method. Now let us set for $n \in \mathbf{N}$

$$Y_n = (y_n^T, y_{n-1}^T, \dots, y_{n-N+1}^T)^T \in \mathbf{C}^{mN}$$

as the vector of the approximations of y_{t_n} at the grid points of Ω_N, that is $y_{n-i} \simeq y_{t_n}(\theta_i)$, $i = 0, \dots, N-1$. We can thus write

$$Y_n = \mathcal{S}_N Y_{n-1}, \quad n \in \mathbf{N} \tag{10}$$

where $\mathcal{S}_N$ is the $mN \times mN$ matrix

$$\mathcal{S}_N = \begin{pmatrix} -\alpha_{k-1}B(hL_0) & \cdots & -\alpha_0 B(hL_0) & \emptyset & \cdots & \emptyset & h\beta_k B(hL_0)L_1 \\ I_m & \cdots & \emptyset & \cdots & \cdots & \cdots & \emptyset \\ \vdots & \ddots & \vdots & & & & \vdots \\ \emptyset & \cdots & I_m & & & & \vdots \\ \vdots & & & \ddots & & & \vdots \\ \vdots & & & & \ddots & & \vdots \\ \emptyset & \cdots & \cdots & \cdots & \cdots & I_m & \emptyset \end{pmatrix}$$

Applying (10) recursively we obtain

$$Y_n = \mathcal{S}_N^n Y_0, \quad n \in \mathbf{N}$$

that is the discretization of (4) at time $t = t_n$. Thus from (8), if $\mu \in \sigma(\mathcal{S}_N)$ we get an approximation λ of the characteristic root λ^*

$$\lambda^* \simeq \lambda = \frac{1}{h} \ln \mu.$$

3.2 The solution operator approach: RK methods For fixed N, N positive integer, let us consider the constant stepsize mesh Ω_N on the interval $[-\tau, 0]$:

$$\text{(11)} \qquad \begin{cases} \Omega_N = \bigcup_{n=1}^{N} \{\theta_n + c_i h \mid i = 1, \ldots, s\} \\ \theta_n = -nh, \quad n = 1, \ldots, N \\ h = \frac{\tau}{N} \\ 0 < c_1 < c_2 < \cdots < c_s = 1 \end{cases}$$

and replace the continuous space X by the discrete space $X_N = (\mathbf{C}^m)^{\Omega_N} = \mathbf{C}^{msN}$.

Applying to (9) an s-stage RK method (A, b, c) such that

- $0 < c_1 < \ldots < c_s = 1$;
- A is invertible;
- $b = (a_{s1}, \ldots, a_{ss})^T$,

(e.g. RADAU-IIA methods satisfy the above conditions) and taking the abscissae $c_1, ..., c_s$ of the RK method as points $c_1, ..., c_s$ in (11) (past values are approximated by past stage values) we obtain

$$\text{(12)} \qquad \begin{cases} Y^{(n+1)} = 1_s \otimes y_n + hA \otimes \left(L_0 Y^{(n+1)} + L_1 Y^{(n+1-N)}\right) \\ y_{n+1} = Y_s^{(n+1)} \end{cases}$$

where $Y^{(k+1)} = \left(Y_1^{(k+1)^T}, ..., Y_s^{(k+1)^T}\right)^T \in \mathbf{C}^{sm}$ is the stage vector at the k-th step and $1_s = (1, 1, \ldots, 1)^T \in \mathbf{R}^s$. Combining equations (12) leads to:

$$Y^{(n+1)} = R(hL_0)(1_s e_s^T \otimes I_m) Y^{(n)} + hR(hL_0)(A \otimes L_1) Y^{(n+1-N)}$$

where

$$R(Z) = (I_s - A \otimes Z)^{-1}$$

is the stability function of the RK method used and $e_s = (0, \ldots, 0, 1)^T \in \mathbf{R}^s$. Now setting

$$[y]_n = (Y^{(n)^T}, Y^{(n-1)^T}, \ldots, Y^{(n+1-N)^T})^T \in X_N$$

we get

$$[y]_{n+1} = \mathcal{S}_N [y]_n, \quad n = 0, 1, 2, ...$$

where $\mathcal{S}_N$ is the $msN \times msN$ matrix:

$$\mathcal{S}_N = \begin{pmatrix} P & \emptyset & \cdots & \emptyset & Q \\ I_{sm} & \emptyset & \cdots & \cdots & \emptyset \\ \emptyset & I_{sm} & \cdots & \cdots & \emptyset \\ \vdots & \vdots & \ddots & & \vdots \\ \emptyset & \emptyset & \cdots & I_{sm} & \emptyset \end{pmatrix}$$

with $P = R(hL_0)(1_s e_s^T \otimes I_m)$ and $Q = hR(hL_0)(A \otimes L_1)$.

The approximation of the characteristic roots are now obtained from the eigenvalues of $\mathcal{S}_N$ in the same way as in the previous section.

3.3 The infinitesimal generator approach: BDF methods The basic idea is to discretize (7) approximating the derivatives of the solution for a suitable choice of points in $[-\tau, 0]$. In order to do this, for fixed N, N positive integer, let us consider the following constant stepsize mesh on the interval $[-\tau, 0]$:

$$\begin{cases} \Omega_N = \{\theta_i = -ih \mid i = 0, \ldots, N\} \\ h = \frac{\tau}{N} \end{cases}$$

and replace the continuous space X by the discrete space $X_N = (\mathbf{C}^m)^{\Omega_N} = \mathbf{C}^{m(1+N)}$.

By using a k-steps ($k \le N$) BDF method and the initial condition outlined in (6) we substitute the derivatives with

$$u_t'(\theta) = \sum_{i=0}^{k} \frac{\alpha_i u_t(\theta - (k-i)h)}{h\beta_k}, \quad \theta \in \Omega_N \setminus \{\theta_0, \ldots, \theta_{k-1}\}$$

and

$$u_t'(\theta) = L_0 u_t(\theta) + L_1 u_t(\theta - Nh), \quad \theta = \theta_0$$

where $u_t(\theta)$ is the approximated solution at the gridpoints of Ω_N at time t. For the remaining $(k-1)$ points of the mesh we set

$$u_t'(\theta_j) = \sum_{i=0}^{k} \frac{\gamma_{ji} u_t(\theta_i)}{h\beta_k}, \quad \theta = \theta_j,\ j = 1, \ldots, k-1$$

where γ_{ji} are suitable coefficients determined by preserving the order k of the method even for the first $(k-1)$ points.

We can thus replace (7) by

$$u_t' = \mathcal{A}_N u_t$$

where $u_t = \left(u_t(\theta_0)^T, u_t(\theta_1)^T, \ldots, u_t(\theta_N)^T\right)^T \in X_N$ and $\mathcal{A}_N$ is the approximated infinitesimal generator given by the $m(1+N) \times m(1+N)$ matrix

$$
(13) \qquad \mathcal{A}_N = \begin{pmatrix}
L_0 & \emptyset & \cdots & \cdots & \cdots & \cdots & \emptyset & L_1 \\
\frac{\gamma_{10} I_m}{h\beta_k} & \frac{\gamma_{11} I_m}{h\beta_k} & \cdots & \cdots & \frac{\gamma_{1k} I_m}{h\beta_k} & \cdots & \cdots & \emptyset \\
\vdots & & \ddots & & \vdots & & & \vdots \\
\frac{\gamma_{k-1 0} I_m}{h\beta_k} & \cdots & \cdots & \ddots & \frac{\gamma_{k-1 k} I_m}{h\beta_k} & \cdots & \cdots & \emptyset \\
\frac{\alpha_0 I_m}{h\beta_k} & \cdots & \cdots & \cdots & \frac{\alpha_k I_m}{h\beta_k} & \cdots & \cdots & \emptyset \\
\vdots & \ddots & & & & \ddots & & \vdots \\
\vdots & & \ddots & & & & \ddots & \vdots \\
\emptyset & \cdots & \cdots & \frac{\alpha_0 I_m}{h\beta_k} & \cdots & \cdots & \cdots & \frac{\alpha_k I_m}{h\beta_k}
\end{pmatrix}
$$

The characteristic roots of (1) are directly approximated by the eigenvalues of $\mathcal{A}_N$.

3.4 The infinitesimal generator approach: RK methods Following the same idea as in the previous section, for fixed N, N positive integer, let us consider the mesh Ω_N on the interval $[-\tau, 0]$:

$$
(14) \qquad \begin{cases}
\Omega_N = \{0\} \cup \bigcup_{n=0}^{N-1} \{\theta_n - c_i h \mid i = 1, \ldots, s\} \\
\theta_n = -nh, \quad n = 0, \ldots, N-1 \\
h = \frac{\tau}{N} \\
0 < c_1 < c_2 < \cdots < c_s = 1
\end{cases}
$$

and replace the continuous space X by the discrete space $X_N = (\mathbf{C}^m)^{\Omega_N} = \mathbf{C}^{m(1+sN)}$.

First to proceed further we establish some notations. For $x \in X_N$ let

$$
x := (x_0^T, [x]_1^T, \ldots, [x]_N^T)^T \in \mathbf{C}^{m(1+sN)}
$$

where

$$
[x]_{n+1} := \left(x(\theta_n - c_1 h)^T, \ldots, x(\theta_n - c_s h)^T\right)^T \in \mathbf{C}^{ms}, \; n = 0, \ldots, N-1
$$

and $x_0 = x(\theta_0) \in \mathbf{C}^m$.

Moreover let us set

$$
x_{n+1} = x(\theta_n - c_s h) = x(\theta_{n+1}) \in \mathbf{C}^m, \; n = 0, \ldots, N-1.
$$

As a discretization of the operator $\mathcal{A}$ in (6) and (7) let us consider the $m(1+sN) \times m(1+sN)$ matrix

$$
(15) \qquad \mathcal{A}_N = \begin{pmatrix} L_0 \quad \emptyset \quad \cdots \quad \emptyset \quad L_1 \\ \mathcal{B}_N \otimes I_m \end{pmatrix}
$$

where:

$$
\mathcal{B}_N = \frac{1}{h} \begin{pmatrix}
w & W & \emptyset & \cdots & \emptyset \\
\emptyset & w & W & \cdots & \emptyset \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
\emptyset & \emptyset & \emptyset & w & W
\end{pmatrix}
$$

is a $sN \times (1+sN)$ matrix with $w := (w_1, \dots w_s)^T \in R^s$,

$$W := \begin{pmatrix} w_{11} & \cdots & w_{1s} \\ \vdots & & \vdots \\ w_{s1} & \cdots & w_{ss} \end{pmatrix} \in \mathbf{R}^{s \times s}$$

and each w is aligned with the last column of W. The matrix $\mathcal{A}_N$ represents a linear operator $X_N \to X_N$ given by

$$(\mathcal{A}_N x)_0 = L_0 x_0 + L_1 x_N \tag{16}$$

and

$$[\mathcal{A}_N x]_{n+1} = \frac{1}{h}\left((w \otimes I_m)x_n + (W \otimes I_m)[x]_{n+1}\right), \; n = 0, \dots, N-1 \tag{17}$$

The equation (16) gives the derivative at the point $\theta_0 = 0$ and corresponds to the initial condition outlined in (6). The equations (17) give approximations of the derivative at the remaining gridpoints of the mesh (14) and correspond to (7). Schemes of discretization can be obtained by s-stage RK methods of the class RADAU-IIA which satisfy the conditions described in section 3.2. In this way we obtain $w = A^{-1}1_s$ and $W = -A^{-1}$ where $1_s = (1, \dots, 1)^T \in \mathbf{R}^s$.

Again the characteristic roots of (1) are directly approximated by the eigenvalues of $\mathcal{A}_N$.

3.5 The infinitesimal generator approach: spectral methods For fixed N, N positive integer, let us consider on the interval $[-\tau, 0]$ a mesh $\Omega_N = \{\theta_i, \quad i = 0, \dots, N\}$ of distinct points and replace the continuous space X by the discrete space $X_N = (\mathbf{C}^m)^{\Omega_N} = \mathbf{C}^{m(1+N)}$.

Let p be the Lagrange interpolant of degree $\leq N$ of $\varphi \in X$ on the mesh Ω_N:

$$p(\theta) = \sum_{j=0}^{N} l_j(\theta)\varphi(\theta_j).$$

We substitute the exact derivatives in (7) with the derivatives of the interpolant p on $\Omega_N \setminus \{\theta_0\}$

$$\varphi'(\theta_i) \simeq \sum_{j=0}^{N} l_j'(\theta)\varphi(\theta_j), \quad i = 1, \dots, N$$

and the exact derivative in 0 by the discretized initial condition outlined in (6)

$$\varphi'(0) \simeq L_0 p(0) + L_1 p(-\tau).$$

Thus the discretized infinitesimal generator $\mathcal{A}_N \in \mathbf{C}^{m(1+N) \times m(1+N)}$ reads

$$\mathcal{A}_N = \begin{pmatrix} L_0 & \emptyset & \cdots & \emptyset & L_1 \\ l_0'(\theta_1) & l_1'(\theta_1) & \cdots & l_{N-1}'(\theta_1) & l_N'(\theta_1) \\ \vdots & \vdots & & \vdots & \vdots \\ l_0'(\theta_N) & l_1'(\theta_N) & \cdots & l_{N-1}'(\theta_N) & l_N'(\theta_N) \end{pmatrix} \tag{18}$$

in the simple case we choose $\theta_0 = 0$ and $\theta_N = -\tau$.

Once more the eigenvalues of $\mathcal{A}_N$ are direct approximations of the characteristic roots of (1).

4 Extension to the multiple and distributed delay cases The infinitesimal generator approach has been extended to the general case (1) for all the methods following the same approach, i.e. applying an opportune quadrature rule to the distributed terms (e.g. compsite Newton-Cotes formulae for BDF methods, RK-based formulae for RK methods and Clenshaw-Curtis formulae for spectral methods) which modifies only the first block-row of (13), (15) and (18) and repeating the single delay approximant matrix for each delay interval $[-\tau_l, -\tau_{l-1}]$, $l = 1, \ldots, k$, in a sparse block-diagonal matrix. Moreover the spectral approach can be alternatively extended approximating the values at the discrete delays and at the quadrature nodes by the same Lagrange interpolant used in section 3.5 on a unique mesh on the whole delay interval $[-\tau, 0]$. Differently from the previous extension, this one leads to a non-sparse matrix. See [3], [2] and [1] for details.

The solution operator approach has not been extended yet to the general case (1).

We remark that for the implementation of the spectral methods we tested both Chebyshev (extremal points) and equispaced meshes.

5 Numerical results Table 1 resumes the relevant characteristics of the *MATLAB* codes which implement all the methods presented in this work.

MATLAB code	**SBDF.m**	**ABDF.m**
theoretical approach	solution operator	infinitesimal generator
numerical method	BDF (order 5)	BDF (order 5)
eigenvalue solver	sparse	sparse
DDE class	single delay	general case
quadrature	-	Newton-Cotes

MATLAB code	**SRK.m**	**ARK.m**
theoretical approach	solution operator	infinitesimal generator
numerical method	RK (order 5)	RK (order 5)
eigenvalue solver	sparse	sparse
DDE class	single delay	general case
quadrature	-	RK based

MATLAB code	**ASPEQ.m**	**ASPCC.m**	**NASPCC.m**
theoretical approach	infinitesimal generator	infinitesimal generator	infinitesimal generator
numerical method	spectral	spectral	piecewise-spectral
eigenvalue solver	standard	standard	sparse
DDE class	general case	general case	general case
quadrature	Clenshaw-Curtis	Clenshaw-Curtis	Clenshaw-Curtis
mesh type	equispaced	Chebyshew	Chebyshew

In this section we present results about the following DDEs:

$$y'(t) = (2 - e^{-2})y(t) + y(t-1) \tag{19}$$

with exact rightmost root $\overline{\lambda} = 2$,

$$\begin{cases} y_1'(t) = -0.5y_1(t) - tanh(y_1(t-1.57)) + tanh(y_2(t-0.2)) \\ y_2'(t) = -0.5y_2(t) + 2.34tanh(y_1(t-0.2)) - tanh(y_2(t-1.57)). \end{cases} \tag{20}$$

linearized around the steady state solution $(y_1^*, y_2^*) = (0,0)$, taken from [7], with rightmost root $\overline{\lambda} = 0.347481725726297$ computed with a tolerance of 10^{-15} and

$$\begin{aligned} y'(t) &= \begin{pmatrix} -3 & 1 \\ -24.646 & -35.430 \end{pmatrix} y(t) + \begin{pmatrix} 1 & 0 \\ 2.356 & -2.004 \end{pmatrix} y(t-1) + \\ &+ \int_{-1}^{-0.5} \begin{pmatrix} 2 & 2.5 \\ 0 & -0.5 \end{pmatrix} y_t(\theta)d\theta + \int_{-0.3}^{-0.1} \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} y_t(\theta)d\theta. \end{aligned} \tag{21}$$

taken from [9], with rightmost root $\overline{\lambda} = -1.246238124592043$ computed with a tolerance of 10^{-15}.

A first comparison of the algorithms is carried out in Figure 1 for the computation of the rightmost root of the single delay equation (19). On the left figure, BDF and RK methods show linear convergence (i.e. $err = \mathcal{O}(N^{-p})$ with $p = 5$ the oder of the method) of the computed root to the exact one, while spectral methods show superlinear convergence (i.e. $err = \mathcal{O}(N^{-N})$). Convergence is proved also for equispaced grids (see [1]). In fact, we are approximating the spectrum of eigenvalues (i.e. the exponential function) and not the solution of the equation. Anyway the test clears out the presence of numerical instability (typical of equispaced mesh), in particular the right figure shows how the (numerically estimated) conditioning number $cond_\infty(\lambda, N)$ relative to the computation of the rightmost eigenvalue of the approximant matrix grows very rapidly for $N \geq 10$. Thus, as well-known in many other numerical applications, equispaced grids are to be avoided while Chebyshev grids are the best-performing (see for example Trefethen, [10]).

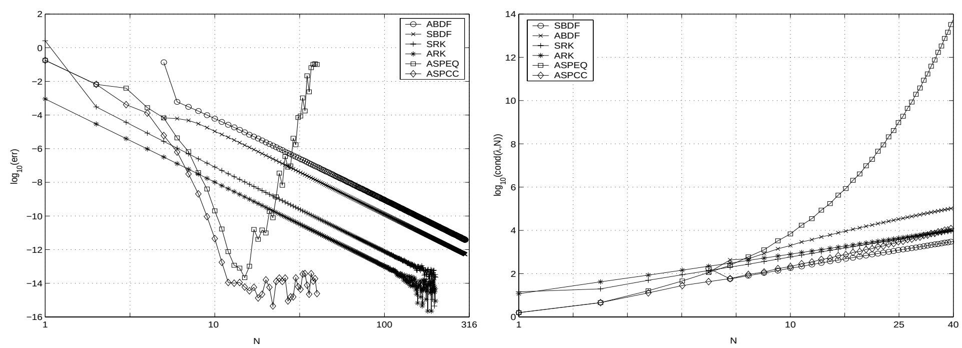

Figure 1: rightmost root error err and conditioning number $cond_\infty(\lambda, N)$ of the rightmost eigenvalue of the approximant matrix *vs* N for system (19).

A deeper comparison is carried out in Figure 2 and 3 for system (20). The infinitesimal generator approach with spectral methods requires the least computational time to match a desired tolerance on the rightmost root (Figure 2, 2nd column). The other approaches are still competitive for lower tolerances. Same conclusions hold in terms of discretization index and approximant matrix dimension (Figure 2, 3rd and 4th columns). The error increases with the modulus of the computed root when more than one root is required (Figure 3, 1st column: the curves correspond in ascending order to $\lambda \simeq -0.081167$, $\lambda \simeq 0.34748$, $\lambda \simeq -0.43412 \pm 1.6275i$ and $\lambda \simeq -0.82062 \pm 5.1118i$) and this accentuates the lag of performance between **ASPCC.m**, **NASPCC.m** and the other algorithms.

Figure 4 analyzes system (21) which involves two distributed terms. Tests on the computation of the rightmost root confirm the results obtained for system (20) apart from **ASPCC.m** for which the presence of integral terms, and consequently the use of a quadrature rule, heavily increases the computational time required to match a given tolerance (Figure 4, 2nd column). This is due to computation of the Lagrange coefficients at the quadrature nodes since these are not necessarily included in the mesh. This is overcame with **NASPCC.m** by using independent mesh and Lagrange interpolant for each delay interval (i.e. piecewise interpolation): in this way quadrature nodes and gridpoints always coincide and any Lagrange coefficient has to be computed.

6 Conclusions In this paper we presented a collection of numerical tests on the computation of characteristic roots for system of DDEs. In particular we briefly described, in the single delay case, two different theoretical approaches (the solution operator integration and

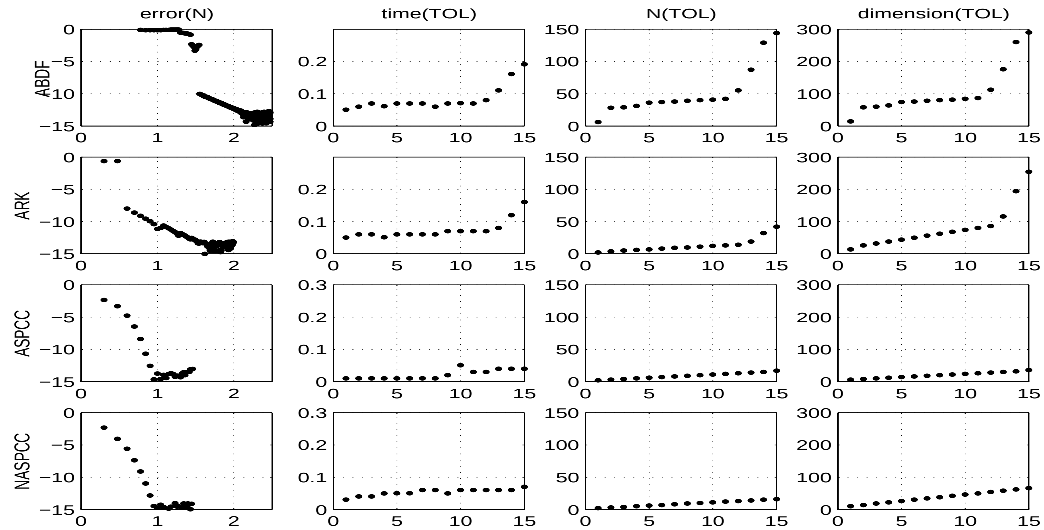

Figure 2: rightmost root computation analysis for system (20).

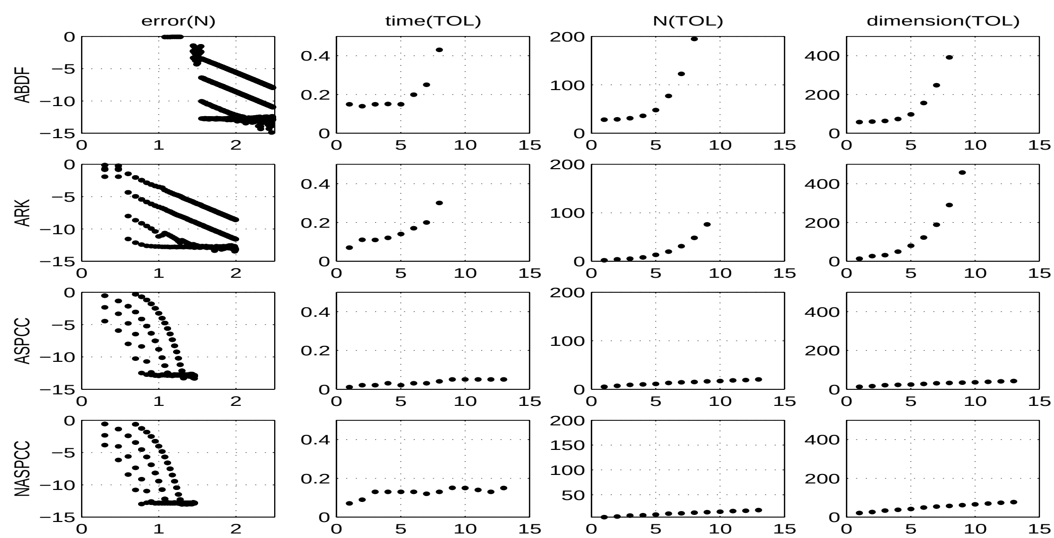

Figure 3: first 6 rightmost roots computation analysis for system (20).

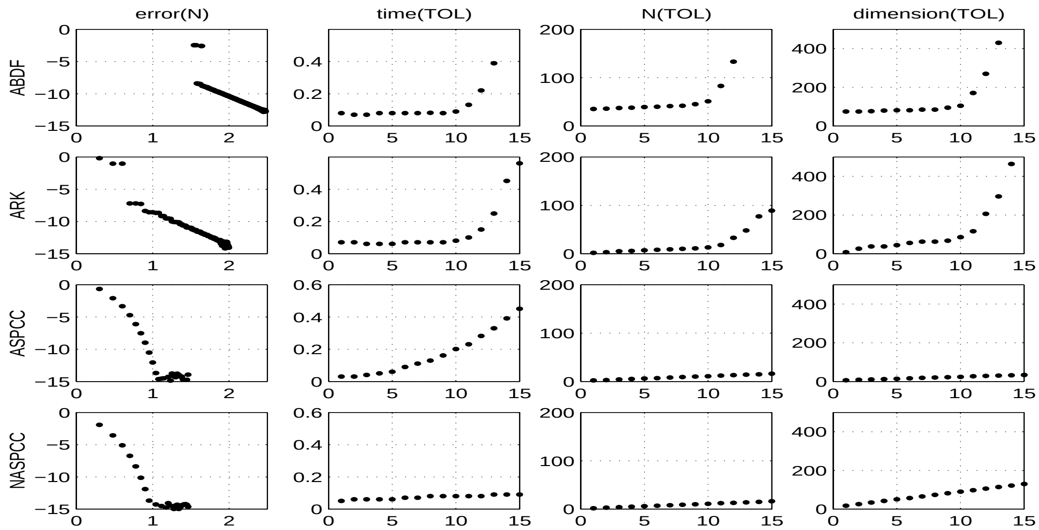

Figure 4: rightmost root computation analysis for system (21).

the infinitesimal generator approximation) implemented by different numerical techniques, namely BDF, RK and spectral methods.

All the resulting algorithms are proved to be convergent via the tests carried out on different systems of DDEs. The codes used in *MATLAB* exploits, wherever possible, the sparseness of the approximant matrices, thus to sensibly reduce the computational time. The results seem to privilege the infinitesimal generator approach implemented with piecewise-spectral methods on Chebyshev points even if BDF and RK methods are still competitive in terms of computational time for low tolerances.

Future work concerning the extensions to neutral DDEs and PDEs with delay will thus focus on the use of spectral methods more than other techniques.

References

[1] Breda, D., Maset, S. and Vermiglio, R. (2002) *Spectral approach for the numerical approximation of characteristic roots for delay differential equations*, work in progress.

[2] Breda, D., Maset, S. and Vermiglio, R. (2002) *Numerical computation of characteristic roots for delay differential equations*, Accepted for publication on IMA J. Numer. Anal..

[3] Breda, D. (2002) *The infinitesimal generator approach for the computation of characteristic roots for delay differential equations using BDF methods*, submitted for publication.

[4] Breda, D. (2002) *Collection of numerical tests on the computation of characteristic roots for delay differential equations*, Research Report RR2/2002, Department of Mathematics and Computer Science, Università di Udine, Italy.

[5] Diekmann, O., van Gils, S.A., Verduyn Lunel, S.M. and Walther, H.O. (1995) *Delay Equations - Functional, Complex and Nonlinear Analysis*, Springer Verlag, AMS series n. 110, New York.

[6] Engelboorghs K., Luzyanina T., Samaey G. (2001) *DDE-BIFTOOL v. 2.00: a Matlab package for bifurcation analysis of delay differential equations*, Report TW330, Department of Computer Science, K. U. Leuven, Belgium.

[7] Engelborghs, K. and Roose, D. (2000) *On Stability of LMS methods and Characteristic Roots of Delay Differential Equations*, accepted for publication on SIAM J. Numer. Anal..

[8] Engelborghs, K. and Roose, D. (1999) *Numerical computation of stability and detection of Hopf bifurcations of steady state solutions of delay differential equations*, Advances in Computational Mathematics, **10** (3-4), 271-289.

[9] Fattouh, A., Sename, O. and Dion, J.M. (2000) *H_∞ controller and observer design for linear systems with point and distributed delays*, proceedings of 2nd IFAC workshop on Linear Time Delay Systems, Ancona, Italy.

[10] Trefethen, Lloyd N. (2000) *Spectral methods in MATLAB*, SIAM, Software-Environmet-Tools series, Philadelphia.

Dottorato in Matematica Computazionale,
Università di Padova
Dipartimento di Matematica e Informatica,
Università di Udine,
Via delle Scienze 208,
I-33100 Udine, Italy
E-mail : dbreda@dimi.uniud.it