



# Critical delays and polynomial eigenvalue problems

Elias Jarlebring

Technische Universität Braunschweig, Institut Computational Mathematics, Pockelsstr 14, 38023 Braunschweig, Germany

## ARTICLE INFO

### Article history:

Received 12 June 2007

### Keywords:

Delay-differential equations  
Quadratic eigenvalue problems  
Critical delays  
Robustness  
Stability

## ABSTRACT

In this work we present a new method to compute the delays of *delay-differential equations* (DDEs), such that the DDE has a purely imaginary eigenvalue. For delay-differential equations with multiple delays, the *critical curves* or *critical surfaces* in delay space (that is, the set of delays where the DDE has a purely imaginary eigenvalue) are parameterized. We show how the method is related to other works in the field by treating the case where the delays are integer multiples of some delay value, i.e., commensurate delays.

The parameterization is done by solving a *quadratic eigenvalue problem* which is constructed from the vectorization of a matrix equation and hence typically of large size. For commensurate delay-differential equations, the corresponding equation is a polynomial eigenvalue problem. As a special case of the proposed method, we find a closed form for a parameterization of the critical surface for the scalar case.

We provide several examples with visualizations where the computation is done with some exploitation of the structure of eigenvalue problems.

© 2008 Elsevier B.V. All rights reserved.

## 1. Introduction

Some phenomena in engineering, physics and biology can be accurately described with *delay-differential equations* (DDEs), sometimes referred to as *time-delay systems* (TDS). The delay in these models typically stem from modelling of phenomena like transmission, transportation and inertia. For instance, in engineering it is often desirable to design a controller for a system where physical limitations make the current state of the system unavailable for measurement. Instead, the feedback in the controller is done with an old state of the system. This is a typical case where a delay-differential equation may be used to model the behavior. See [23] for more examples. In this paper we consider linear, time-invariant,  $n$ -dimensional delay-differential equations with  $m$  delays,

$$\begin{cases} \dot{x}(t) = A_0 x(t) + \sum_{k=1}^m A_k x(t - h_k), & t > 0 \\ x(t) = \varphi(t), & t \in [-h_m, 0] \end{cases} \quad (1)$$

with  $x: [-h_m, \infty) \mapsto \mathbb{R}^n$  and  $h_k \in \mathbb{R}_+$ ,  $A_k \in \mathbb{R}^{n \times n}$  for  $k = 0, \dots, m$ .

For most applications it is desirable that the solution is stable independent of the initial condition, i.e., that the solution goes asymptotically to zero when time goes to infinity. This and many other property can be determined from the solutions of the characteristic equation. The *characteristic equation* of (1) is

$$\mathbb{M}(s)v := \left( -sI_n + A_0 + \sum_{k=1}^m A_k e^{-h_k s} \right) v = 0, \quad \|v\| = 1, \quad (2)$$

where  $v \in \mathbb{C}^n$  is called *eigenvector* and  $s \in \mathbb{C}$  an *eigenvalue*. The set of all eigenvalues is called the *spectrum*.

E-mail address: [e.jarlebring@tu-bs.de](mailto:e.jarlebring@tu-bs.de).

Similar to ordinary differential equations (ODEs), a DDE is exponentially stable if and only if all eigenvalues lie in the open left complex half-plane, because the only clustering point of the real part is minus infinity. An essential difference is that, unlike ODEs, the spectrum of DDEs contains a countably infinite number of eigenvalues. Fortunately, it can be proven (cf. [15]) that there are only a finite number of eigenvalues to the right of any vertical line in the complex plane, e.g., there are only finite number of unstable eigenvalues.

It is of particular interest to determine for what choices of the delays  $h_1, \dots, h_m$  the DDE (1) is stable. This set of  $h_1, \dots, h_m$  is referred to as the *stability region* in *delay-parameter space*. From continuity it is clear that, at the boundary of any stability region in the delay-parameter space, the DDE has at least one purely imaginary eigenvalue. These critical delay values (single delay), critical curves (two delays) or critical surfaces (more delays) are hence important for a complete stability analysis.<sup>1</sup> In this work we focus on the computation of critical curves and surfaces.

In Section 2.3 we present a new method to parameterize such curves and surfaces. An important step of the method is to compute the eigenvalues of a large *quadratic eigenvalue problem*, cf. [33], such that the eigenvalue is on the unit circle. The matrices of the eigenvalue problem are of size  $n^2 \times n^2$  and hence large even for DDEs of moderate size. The size of the matrices will cause a computational bottleneck and we therefore make some initial notes on how an implementation can make use of the structure of the problem of the corresponding linearized eigenvalue problem. In Section 2.4 we provide a new interpretation of the results in [6], where the *commensurate case* is treated. The critical delays for a system with *commensurate delays* can be computed from the eigenvalues of a polynomial eigenvalue problem of degree  $2m$ .

The large number of approaches in the literature yielding delay-dependent stability conditions can be classified into two main branches. Most results are either based on the construction of a Lyapunov–Krasovskii functional, or (as here) based on a discussion of the imaginary eigenvalues.

Both types of approaches have advantages and disadvantages. For instance, the methods in [9,24] and the methods summarized in [11] are examples where a Lyapunov–Krasovskii functional is used to construct sufficient but conservative delay-dependent stability conditions formulated as *linear matrix inequalities* (LMIs). An advantage of these types of approaches is that the conditions can be elegantly formulated with LMIs which allow the automatic application in engineering software. A disadvantage is that the results can be conservative and may impose unnecessarily strong constraints. Moreover, the treatment of LMIs of large dimension may be computationally cumbersome.

The approaches based on imaginary eigenvalues typically yield exact results but cannot be compactly formulated. Early works (and some more recent works) with a discussion of the imaginary eigenvalues were limited to special cases. For instance, the trigonometric conditions in [26, Chapter III] are for scalar problems with only two delays, the conditions in [7] for second-order DDEs with a single delay, the geometric characterization in [14] for first-order (scalar) systems with two delays, the discussion using the quasipolynomial in [3] for single-delay DDEs where the coefficients are dependent on the delay, the analysis of two-delay DDEs of a special form in [12] (further discussed in Example 1), the analysis of third-order DDEs with two delays by Sipahi, et al. in [30] based on the earlier methods in [32,27,16], the analysis based on a Pontryagin's results for quasipolynomials for second-order (scalar) DDEs in [5], the explicit trigonometric analysis of scalar two-delay DDEs in [2]. From this large (but incomplete) list of works, we conclude that an analysis based on imaginary eigenvalues is indeed accepted as a natural way to construct stability conditions.

Recent focus in this field is towards neutral DDEs, distributed DDEs and conditions which can be efficiently and accurately computed on a computer. The importance of the possibility to apply the conditions using a computer should not be underestimated. In fact, the conditions from most of the methods mentioned above are implicitly or explicitly formulated using the coefficients in the characteristic polynomial with exponential terms (sometimes referred to as a quasipolynomial). It is well known from numerical analysis of eigenvalue problem, that this (nominal) representation of the characteristic equation is not (numerically) stable, i.e., the eigenvalues can be (and are often) very sensitive with respect to perturbations (e.g. rounding errors) in the coefficients. This is important, since it is not possible to represent a polynomial on a computer using its coefficients without introducing small errors. For this reason, the methods mentioned above applied to systems of order (say) 4 or more typically do not give numerical results which can be used in practice. This motivates the recent works on imaginary eigenvalues using a formulation of the matrices in the characteristic equation which is generally believed to scale better with the dimension of the DDE. The following works represent the class of conditions formulated as generalized eigenvalue problems, [6] for commensurate systems and derivative works [10,25,19] for single-delay neutral systems. In the conditions of these matrix-pencil methods, the eigenvalues of a large generalized eigenvalue problem must be computed. Similar to the method discussed here, the matrices are constructed using the Kronecker product.

We also mention the related work in [8] which is partially based on reason with matrices, but also uses the coefficients in the quasipolynomial. The method presented here has similarities with this work.

Finally, we note that there are methods to numerically determine parts of the spectra, e.g. using a *linear multistep discretization* of the solution operator [34] or discretization of the infinitesimal generator of the semigroup corresponding to the DDE [4]. These methods can be applied for a grid of points in delay space. If the grid is fine enough, a numerical stability condition is computed.

Apart from the standard works in the field [11,23], we refer the reader to the references in the introduction of [18] for a well-balanced overview of modern delay-dependent stability conditions.

<sup>1</sup> The critical curves (surfaces) are called *offspring curves* and *kernel curves* in [30], *Hopf bifurcation curves (surfaces)* in [13] and *crossing delays (curves)* in [22] and [12].

## 2. Results

In order to clearly state the results we start by presenting a motivating example in Section 2.1. The example describes the problem as well as the idea behind the method.

The main result consists of the method to parameterize the critical curves (surfaces) in Section 2.3 and some further interpretation and relation to other publications for the case that the delays are commensurate in Section 2.4. The computationally dominating part of both of the methods is the determination of unit eigenvalues of *quadratic* or *polynomial eigenvalue problems* of large dimension.

### 2.1. Motivation

We now describe the general idea behind the method presented in Section 2.3. We do this by first considering the scalar two-delay case and describe how the derivation must be changed to hold for multiple dimensions and multiple delays.

We wish to find purely imaginary eigenvalues, say  $s = i\omega$  with corresponding eigenvector  $v$ , i.e.,  $\mathbb{M}(i\omega)v = 0$ . In order to clearly motivate the results we discuss the two-delay case and generalize and formalize the discussion in the following subsections. The characteristic equation for a DDE with two delays is

$$\mathbb{M}(i\omega)v = (-i\omega I + A_0 + A_1 e^{-ih_1\omega} + A_2 e^{-ih_2\omega})v = 0.$$

Except for some special cases, the points (in delay space) of interest, i.e., points which causes the DDE to have imaginary eigenvalues, correspond to curves. We wish to parameterize these *critical curves*, and pick  $\varphi := h_1\omega$  as a free parameter. By treating  $\varphi$  as a known value we form a condition on  $h_2$  and  $\omega$  (such that we can also compute  $h_1$ ). We will see later that this choice of free parameter is valid for most cases. The characteristic equation is  $2\pi$ -periodic in  $\varphi$ , and it will be enough to let  $\varphi$  run the whole span  $[-\pi, \pi]$ , i.e. for each choice of  $\varphi$  in this interval we will find some points on the critical curves and if we let  $\varphi$  run the whole interval we will find all critical points. The characteristic equation is such that we wish to find all  $\omega \in \mathbb{R}$  and  $z \in \partial D$  such that

$$\mathbb{M}(i\omega)v = (-i\omega I + A_0 + A_1 e^{-i\varphi} + A_2 z)v = 0, \quad v \neq 0 \quad (3)$$

where  $\partial D$  denotes the unit circle. If we first consider the scalar case, i.e.,  $A_0 = a_0, A_1 = a_1, A_2 = a_2 \in \mathbb{R}$  the equation corresponds to two scalar conditions (say real and imaginary parts) and we can eliminate either  $\omega$  or  $z$ . In the approach presented here, we eliminate  $\omega$  by forming the sum of (3) and its complex conjugate, i.e.,

$$0 = 2a_0 + a_1 e^{-i\varphi} + a_2 \bar{z} + a_1 e^{i\varphi} + a_2 z = a_2 \bar{z} + 2a_0 + 2a_1 \cos(\varphi) + a_2 z.$$

Multiplying with  $z$  yields the quadratic equation,

$$a_2 z^2 + 2z(a_0 + a_1 \cos(\varphi)) + a_2 = 0,$$

since  $z\bar{z} = 1$ . It has the two solutions

$$z = \frac{-(a_0 + a_1 \cos(\varphi)) \pm i\sqrt{a_2^2 - (a_0 + a_1 \cos(\varphi))^2}}{a_2},$$

assuming  $a_2 \neq 0$ . We can now compute  $\omega$  by inserting  $z$  into (3) and rearrange the terms, i.e.,

$$i\omega = a_0 + a_1 e^{-i\varphi} + a_2 z = i \left( -a_1 \sin(\varphi) \pm \sqrt{a_2^2 - (a_0 + a_1 \cos(\varphi))^2} \right).$$

Since  $z = e^{-ih_2\omega}$  and  $\varphi = h_1\omega$ , a parameterization of the critical curves is

$$\bar{h}(\varphi) = \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} \frac{\varphi + 2p\pi}{-a_1 \sin(\varphi) \pm \sqrt{a_2^2 - (a_0 + a_1 \cos(\varphi))^2}} \\ \frac{-\text{Arg}z + 2q\pi}{-a_1 \sin(\varphi) \pm \sqrt{a_2^2 - (a_0 + a_1 \cos(\varphi))^2}} \end{pmatrix}, \quad (4)$$

where  $\text{Arg}z = \pm \text{sign}(a_2) \arccos\left(-\frac{a_0 + a_1 \cos(\varphi)}{a_2}\right)$  for any  $p, q \in \mathbb{Z}$  and  $\varphi \in [-\pi, \pi]$ . Note that the signs must be matched, i.e., for each choice of the free parameter  $\varphi$  and branches  $p$  and  $q$ , the parameterization has two critical delays.

There are other approaches based on other parameterizations. For instance, the analysis of Gu et al. [12] can be seen as a parameterization with  $\omega \in \mathbb{R}$  as free parameter. They reach the result that if the characteristic equation can be rewritten to the form

$$1 + a_1(s)e^{-h_1 s} + a_2(s)e^{-h_2 s} = 0,$$

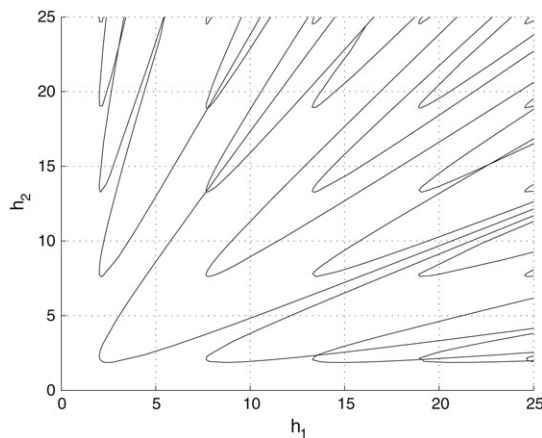


Fig. 1. Critical curves for Example 1.

where  $a_1$  and  $a_2$  are rational functions, a parameterization of the critical curves are given by

$$h_1 = \frac{\text{Arg}(a_1(i\omega)) + (2p-1)\pi \pm \theta_1}{\omega}, \quad \theta_1 = \arccos\left(\frac{1 + |a_1(i\omega)|^2 - |a_2(i\omega)|^2}{2|a_1(i\omega)|}\right),$$

$$h_2 = \frac{\text{Arg}(a_2(i\omega)) + (2q-1)\pi \mp \theta_2}{\omega}, \quad \theta_2 = \arccos\left(\frac{1 + |a_2(i\omega)|^2 - |a_1(i\omega)|^2}{2|a_2(i\omega)|}\right).$$

We also note that the different types of possible critical curves and other properties are classified in [14].

**Example 1** (Different Parameterizations). In order to show the difference between (4) and the parameterization in [12], we construct the parameterizations of the critical curves for the case that  $a_0 = a_1 = -1$ ,  $a_2 = -\frac{1}{2}$ . According to (4), the critical curves are given by,

$$h(\varphi) = \begin{pmatrix} \frac{\varphi + 2p\pi}{\sin(\varphi) \pm \frac{1}{2}\sqrt{5 + 8\cos(\varphi) + 4\cos^2(\varphi)} \pm \arccos(-2 - 2\cos(\varphi)) + 2q\pi} \\ \frac{\varphi + 2q\pi}{\sin(\varphi) \pm \frac{1}{2}\sqrt{5 + 8\cos(\varphi) + 4\cos^2(\varphi)} \mp \arccos(-2 - 2\cos(\varphi)) + 2p\pi} \end{pmatrix}.$$

In this example we can find a slightly more elegant parameterization by letting  $x = \cos(\varphi)$ , i.e.,

$$h(x) = \begin{pmatrix} \frac{\arccos(x) + 2p\pi}{-\text{sign}(x)\sqrt{1-x^2} \pm \sqrt{0.25 - (1+x)^2} \pm \arccos(-2-2x) + 2q\pi} \\ \frac{\arccos(x) + 2q\pi}{-\text{sign}(x)\sqrt{1-x^2} \pm \sqrt{0.25 - (1+x)^2} \mp \arccos(-2-2x) + 2p\pi} \end{pmatrix}, \quad x \in [-1, 1].$$

The critical curves are shown in Fig. 1. The minimum of the 2-norm, i.e. the 2-norm stability delay radius, is  $\|h(x)\|_2 \approx 2.896$  and taken at  $x \approx -0.7012$  where  $\omega \approx 1.1139$ ,  $h_1 \approx 2.1078$  and  $h_2 \approx 1.9853$ .

In the context of [12],  $a_1(i\omega) = \frac{1}{1+i\omega}$ ,  $a_2(i\omega) = \frac{1}{2(1+i\omega)}$  and the parameterization is given by

$$h_1 = \frac{-\text{Arg}(1+i\omega) + (2p-1)\pi \pm \theta_1(\omega)}{\omega}$$

$$= \frac{1}{\omega} \left( -\text{atan}(\omega) \pm \arccos\left(\frac{1}{2}\sqrt{1+\omega^2} + \frac{3}{8\sqrt{1+\omega^2}}\right) + (2p-1)\pi \right)$$

$$h_2 = \frac{-\text{Arg}(1+i\omega) + (2q-1)\pi \mp \theta_2(\omega)}{\omega}$$

$$= \frac{1}{\omega} \left( -\text{atan}(\omega) \mp \arccos\left(\sqrt{1+\omega^2} - \frac{3}{4\sqrt{1+\omega^2}}\right) + (2q-1)\pi \right),$$

which represent the same set of curves, but has a surprisingly small obvious similarity with the other parameterization (4).

We now wish to do an analysis similar to the derivation of (4) for the general case where  $A_0$ ,  $A_1$  and  $A_2$  are not scalar. It turns out that a straightforward generalization does not yield such explicit results. The reason is that the elimination of  $\omega$  must be done in a slightly different way. The first step in the discussion above involves taking the sum of (3) and its complex

conjugate. Since  $v$  is typically a complex vector, the sum of (3) and its complex conjugate (3) will not eliminate  $\omega$ . Taking the sum of the (3) and its complex conjugate transpose is of course also not possible as the dimensions do not fit. Instead we take the sum of  $\mathbb{M}(s)vv^*$  and  $vv^*\mathbb{M}(s)^*$ , where  $A^*$  denotes the complex conjugate transpose of  $A$ . We believe that this is a natural way to generalize the elimination of  $\omega$ . Clearly,

$$\begin{aligned} 0 &= \mathbb{M}(s)vv^* + vv^*\mathbb{M}(s)^* \\ &= A_2vv^*z + ((A_0 + A_1e^{-i\varphi})vv^* + vv^*(A_0^T + A_1^Te^{i\varphi})) + vv^*A_2^T\bar{z}, \end{aligned}$$

which is a matrix equation. This equation can be rewritten into a *quadratic eigenvalue problem* by vectorization, i.e., stacking the columns of the matrix on top of each other. Quadratic eigenvalue problems can be rewritten into a generalized eigenvalue problem using a so-called *companion linearization* (c.f. [21]). As we will see later, if  $n$  is not large, there are numerical methods to find all  $z$ . We can then compute the corresponding  $\omega$  and the critical delays similar to the way done for the scalar case.

Even though the motivation above is simple, it is not clear that all steps involved are reversible, i.e., there is no convincing argument to ensure that we get all critical delays. For that reason, we formalize the discussion in the sections that follow. At the same time we generalize the method to an arbitrary number of delays. The generalization is such that if we have  $m$  delays we have  $m - 1$  free parameters, and need to solve a quadratic eigenvalue problem for each choice of the parameters. In order to compare the method with related approaches we also discuss the case where the delays are (fixed) integer multiples of the first delay, i.e., commensurate delays.

## 2.2. Main results

Our main goal is to formalize and generalize the discussion in the previous section, and to see how the constructed method relates to approaches in the literature. In particular, we wish to provide further understanding to the method in [6] where a matrix-pencil method is presented for the *commensurate case*, i.e., the case where the delays are integer multiples of the first delay. We do this by first, in a general setting, introducing an operator  $\mathbb{L}$ . For this operator we can formulate an equivalence theorem with the characteristic equation. One reason for formulating such a theorem is that the results in [6] are easily interpreted from the conditions in the equivalence theorem. Another reason is that from the conditions we can construct a new method (similar to the motivation) to compute the critical delays for the case that the delays are not necessarily commensurate. Hence, both results can be interpreted in terms of the conditions in the equivalence theorem.

**Theorem 2.** Given  $s \in \mathbb{C}$  and  $v \in \mathbb{C}^n$ ,  $v^*v = 1$  the following statements are equivalent.

$$\mathbb{M}(s)v = 0 \quad (5)$$

$$\mathbb{L}(vv^*, s) = 0 \wedge v^*\mathbb{M}(s)v = 0 \quad (6)$$

where

$$\begin{aligned} \mathbb{L}(X, s) &:= (\mathbb{M}(s))X + X(\mathbb{M}(s)^*) \\ &= \sum_{k=0}^m (A_kXe^{-h_k s} + XA_k^Te^{-h_k \bar{s}}) - 2X\operatorname{Re} s, \end{aligned} \quad (7)$$

and  $h_0 = 0$  for notational convenience.

**Proof.** The forward implication is trivial from definitions, i.e., (2) and (7). The backward implication, i.e., (6)  $\Rightarrow$  (5), is clear from the following equality.

$$\mathbb{M}(s)v = \mathbb{L}(vv^*, s)v - vv^*\mathbb{M}(s)^*v. \quad \square$$

The reason why we construct  $\mathbb{L}$  in this way, is that similar to the motivation, for the critical case, i.e.,  $s = i\omega \in i\mathbb{R}$ ,  $s = i\omega$  only appears in the exponential terms of the expression.

In the two following sections we rephrase the equivalence theorem such that we get a method to compute the critical delays. Because of the equivalence theorem the formulation will not miss any critical delays.

## 2.3. Free delays

In the motivation we saw that for two delays we needed to parameterize a *curve* in delay space. In general, if we have  $m$  delays we need to parameterize the  $(m - 1)$ -dimensional (hyper-)surface. We do that by letting  $z = e^{-ih_m\omega}$  and represent the other exponential terms by free variables, i.e.,  $\varphi_k = h_k\omega$ ,  $k = 1, \dots, m - 1$ . This allows us to rewrite the main expression of Theorem 2 to a quadratic eigenvalue problem which can be solved by transforming it to an eigenvalue problem using a *companion linearization*.

For the scalar case, the quadratic eigenvalue problem can be solved explicitly, and we can form an explicit parameterization of the critical surfaces. We also verify the results by comparing it to classical results for the single delay, scalar delay-differential equations.

The following theorem follows from the substitution  $z = e^{-ih_m\omega}$ ,  $\varphi_k = h_k\omega$ ,  $k = 1, \dots, m-1$  and Theorem 2.

**Theorem 3.** The critical delays are given by  $h_m = \frac{-\text{Arg}z + 2p_m\pi}{\omega}$ ,  $h_k = \frac{\varphi_k + 2p_k\pi}{\omega}$ ,  $k = 1, \dots, m-1$ , where  $p_k \in \mathbb{Z}$ ,  $k = 1, \dots, m$ ,  $\varphi_k \in [-\pi, \pi]$ ,  $k = 1, \dots, m-1$ ,  $z \in \partial D$ ,  $\omega \in \mathbb{R}$  such that

$$z^2 A_m v v^* + z \left( \sum_{k=0}^{m-1} A_k v v^* e^{-i\varphi_k} + v v^* A_k^T e^{i\varphi_k} \right) + v v^* A_m^T = 0, \quad (8)$$

$$\omega = -iv^* \left( A_m z + \sum_{k=0}^{m-1} A_k e^{-i\varphi_k} \right) v, \quad (9)$$

where  $\varphi_0 = 0$  for notational convenience.

Apart from singular cases (which will be discussed later), this theorem generates a parameterization of the critical curves, with  $\varphi_1, \dots, \varphi_{m-1}$  as free variables. It is clear that if we can find solutions  $z, v$  of (8),  $\omega$  and critical delays can be computed for a specific choice of  $\varphi_k$ ,  $k = 1, \dots, m-1$ . Hence, the difficulty of applying this theorem is solving (8). We solve this by vectorizing and rewriting the quadratic equation to a generalized eigenvalue problem.

The vectorized version of (8), i.e., the same equation but the columns of the matrix condition stacked on each other, is the equation

$$\left( z^2 I \otimes A_m + z \sum_{k=0}^{m-1} L_k(\varphi_k) + A_m \otimes I \right) u = 0, \quad (10)$$

where  $u = v \otimes \bar{v}$  is the vectorization of  $vv^*$  and  $L_k(\varphi_k) := I \otimes A_k e^{-i\varphi_k} + A_k \otimes I e^{i\varphi_k}$ , and  $\otimes$  the Kronecker product with the usual meaning. Problems of the type  $(Mz^2 + Gz + K)v = 0$ , e.g., (10), are known in the literature as *quadratic eigenvalue problems* (cf. [33]). They are typically solved by transformation into first-order form, normally using a so-called *companion linearization*, e.g., here

$$\begin{pmatrix} 0 & I \\ I \otimes A_m & \sum_{k=0}^{m-1} L_k(\varphi_k) \end{pmatrix} \begin{pmatrix} u \\ zu \end{pmatrix} = z \begin{pmatrix} I & 0 \\ 0 & -A_m \otimes I \end{pmatrix} \begin{pmatrix} u \\ zu \end{pmatrix}. \quad (11)$$

We can hence find the solutions of (8) by finding the eigenvalues corresponding to (11), to which standard methods for generalized eigenvalue problem can be applied. There are also other numerical methods to solve quadratic eigenvalue problems without linearizing it, e.g., *Jacobi–Davidson* [31] and *second-order Arnoldi* [1]. In this paper we focus on the companion linearization (11).

With these methods it possible (unless  $n$  is large) to find the eigenpairs of (11) to sufficient accuracy on a computer. Because it is not possible (in general) to solve eigenvalue problems without errors, we will not get exactly a vector corresponding to a rank-one matrix. However, if the eigenvalue is isolated we can find an approximation close to the rank-one matrix. We pick the approximation by choosing the rank-one matrix corresponding to the singular value which is not close to zero. All but one singular value should be close to zero, because the numerical rank is one. The accuracy of this choice of the approximation can be characterized by the following simple result.

**Lemma 4.** Let  $(z, x)$  be an eigenpair of the quadratic eigenvalue problem  $(z^2 M + zG + K)x = 0$  and  $x = v \otimes \bar{v}$ ,  $\|x\| = 1$ ,  $|z| = 1$ . Say  $\tilde{x} = x + y = u \otimes \bar{u} + q$  is an approximation of  $x$ , then the sine of the angle between the approximation  $u$  and the vector  $v$  is bounded by

$$|\sin(u, v)| \leq \sqrt{\|y\| (\|M + I\| + \|G\| + \|K\|) + \|q\|}.$$

**Proof.** Multiplying the characteristic equation from the left with  $\tilde{x}^*$  and adding  $z^2 \tilde{x}^* x$  to both sides, yields

$$\tilde{x}^* x z^2 = \tilde{x}^* (z^2 (M + I) + zG + K)x.$$

We exploit that  $\tilde{x}^* x = (u \otimes \bar{u} + q)^* (v \otimes \bar{v}) = u^* v \bar{u}^* \bar{v} + q^* x$  from which we deduce that

$$\begin{aligned} z^2 |u^* v|^2 &= z^2 (u^* v \bar{u}^* \bar{v}) = \tilde{x}^* (z^2 (M + I) + zG + K)x - z^2 q^* x \\ &= x^* (z^2 (M + I) + zG + K)x + y^* (z^2 (M + I) + zG + K)x - z^2 q^* x \\ &= z^2 + y^* (z^2 (M + I) + zG + K)x - z^2 q^* x, \end{aligned}$$

i.e.,

$$|u^* v|^2 = 1 + y^* ((M + I) + z^{-1}G + Kz^{-2})x - q^* x$$



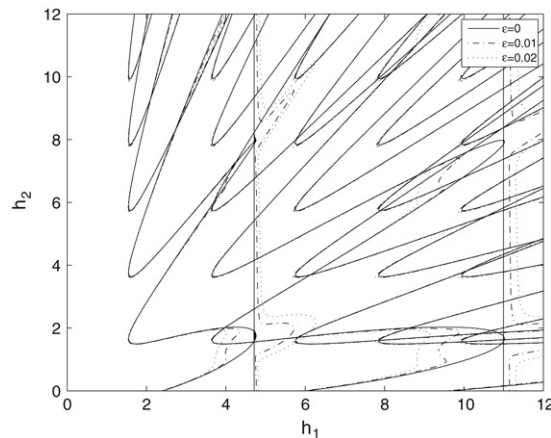


Fig. 2. Critical curves for Example 5.

and

$$|u^*v|^2 \geq 1 - \|y\| (\|M + I\| + \|G\| + \|K\|) - \|q\|,$$

where we used that  $|z| = 1$ . Finally, the result follows from the fact that  $|\sin(u, v)| = \sqrt{1 - |u^*v|^2}$ .  $\square$

In rough terms, the result is that if the error of the eigenvector, denoted by  $y$ , is small and the distance to the Hermitian rank-one matrix  $q$  is small, then the angle between  $v$  and its approximation  $u$  is small. The lemma also indicates the (unfortunate) fact that if the error of the eigenvector is  $\epsilon_{mach}$  the accuracy of the approximation  $u$  is essentially  $\sqrt{\epsilon_{mach}}$ .

If we have multiple eigenvalues, the invariant subspace must be sought for rank-one matrices. This typically unusual case will not be treated in detail.

Moreover, we are only interested in eigenvalues  $z$  on the unit circle. Because of rounding errors, we classify an eigenvalue as a unit-eigenvalue if its modulus differs from 1 less than some tolerance, say  $\sqrt{\epsilon_{mach}}$ .

It is also worth noting that (10) is a quadratic eigenvalue problem and can have (the so-called) *infinite eigenvalues*. For instance, if  $A_m = 0$  the equation is independent of  $z \in \mathbb{C}$ , i.e., if the linear term is zero it is fulfilled for any  $z \in \partial D$ . The interpretation of the theorem is then that the criticality is independent of  $h_m$ , and one of the free parameters is no longer free. Since this can only occur if  $A_m$  is singular, the problems of infinite eigenvalues can be avoided by reordering of the matrices (assuming at least one matrix is nonsingular). We characterize the phenomena of infinite eigenvalues with the following example.

**Example 5 (Singular Case).** Consider the DDE with

$$A_0 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad A_1 = \begin{pmatrix} 2 & \varepsilon \\ 3 & 1 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}.$$

With this example we wish to show how an infinite eigenvalue of (10) can be interpreted, and how this appears in Theorem 3. For this particular example it is actually possible to circumvent the problem with the infinite eigenvalue simply by switching  $A_1$  and  $A_2$ . Here, we will not approach the problem that way, because that type of reordering of the matrices is not always possible.

If (10) has an infinite eigenvalue, it must have a corresponding eigenvector such that the quadratic term disappears, i.e., here  $u = (0, 1)^T \otimes (0, 1)^T = (0, 0, 0, 1)^T$  is the only eigenvector of the correct form. For this  $v$ , the quadratic and the constant term in (8) are zero, and (9) is independent of  $z$ . Since  $z \neq 0$ ,  $v$  is a valid solution of (8) only if

$$A_1 v v^* e^{-i\varphi_1} + v v^* A_1^T e^{i\varphi_1} = 0. \quad (12)$$

This means that the only choice of  $\varphi_1$  which generates a critical delay (corresponding to this  $v$ ) is when (12) is fulfilled. Here (12) and hence (8) is fulfilled only if  $\varepsilon = 0$  and for  $\varphi_1 = -\frac{\pi}{2} + 2p\pi$  with the corresponding critical curve  $h_1 = -\frac{\pi}{2} + 2p\pi$  (for any  $h_2$ ). The critical curves are plotted for three choices of  $\varepsilon$  in Fig. 2. From the figure it is easy to identify that for  $\varepsilon = 0$  there are a vertical critical curves, i.e., curves which are independent of  $h_2$  corresponding to the contribution of the infinite eigenvalue. If  $\varepsilon > 0$  there are no vertical lines, no infinite eigenvalues and the critical curves are characterized by the finite eigenvalues.

For the scalar case, the quadratic eigenvalue problem reduces to a quadratic equation. The theorem can then be simplified to

**Corollary 6.** For the case that  $A_k = a_k \in \mathbb{R}$ ,  $k = 0, \dots, m$ , the critical delays are given by

$$h_k = \frac{\varphi_k + 2p_k\pi}{\omega}, \quad k = 1, \dots, m-1$$

$$\mp \text{sign}(a_m) \arccos \left( \frac{\sum_{k=0}^m a_k \cos(\varphi_k)}{a_m} \right) + 2p_m\pi$$

$$h_m = \frac{\omega}{\omega},$$

$$\omega = -\sum_{k=0}^m a_k \sin(\varphi_k) \pm \sqrt{a_m^2 - \left( \sum_{k=0}^m a_k \cos(\varphi_k) \right)^2}$$

where  $p_k \in \mathbb{Z}$ ,  $k = 1 \dots m$ ,  $\varphi_k \in [-\pi, \pi]$ ,  $k = 1, \dots, m-1$ .

It is worth noting that the formulas in the literature for the scalar single-delay case normally assume  $a_1 \leq -|a_0|$ , because otherwise the DDE is (un)stable independent of delay or the corresponding delay-free DDE is not stable. From Corollary 6, the critical delays for the single-delay scalar DDE are given by

$$h = \frac{-\text{sign}(a_1) \arccos \left( -\frac{a_0}{a_1} \right) + 2p\pi}{\sqrt{a_1^2 - a_0^2}},$$

which reduces to classical results if  $a_1 < 0$ , cf. [7] or [23, Section 3.4.1].

#### 2.4. Commensurate delays

For the case that the delays  $h_k$  are integer multiples of the smallest delay, say  $h$ , Theorem 2 can be restated such that it is very similar to the results by Chen, Gu and Nett in [6]. For  $h_k = n_k h$ , the problem is to find the critical delay  $h$  such that the choice of the delays in the direction  $\bar{h} = (h_1, h_2, \dots, h_m) = (n_1 h, n_2 h, \dots, n_m h)$ , where  $n_1, \dots, n_m \in \mathbb{N}$ , generates a purely imaginary eigenvalue. In other words, we fix a “rational” direction in delay space with  $(n_1, \dots, n_m)$  and search for critical delays along this direction.

The following theorem follows from Theorem 2 by letting  $z = e^{-ih\omega}$ .

**Theorem 7.** The critical delays  $h \in \mathbb{R}_+$  corresponding to the direction in delay space defined by

$$\bar{h} = (h_1, \dots, h_m) = (hn_1, hn_2, \dots, hn_m),$$

are given by

$$h = \frac{-\text{Arg} z + 2p\pi}{\omega},$$

for any  $p \in \mathbb{Z}$  and for  $v \in \mathbb{C}^n$ ,  $v^* v = 1$ ,  $\omega \in \mathbb{R}$ ,  $z \in \partial D$  fulfilling

$$\sum_{k=0}^m (A_k v v^* z^{n_k} + v v^* A_k^T z^{-n_k}) = 0, \quad (13)$$

and

$$i\omega = v^* \left( \sum_{k=0}^m A_k z^{n_k} \right) v,$$

where  $n_0 = 0$  for notational convenience.

Analogously to the previous section, if it is possible to find the solutions  $z$  and  $v$  of (13) the other expressions explicitly yield  $\omega$  and  $h$ . Without loss of generality we let  $n_m = \max_{k \in [1, \dots, m]} n_k$ . After vectorizing the matrix equation (13) we find that

$$\sum_{k=0}^m (I \otimes A_k z^{n_m+n_k} + A_k \otimes I z^{n_m-n_k}) u = 0 \quad (14)$$

where  $u \in \mathbb{C}^{n^2}$  is the vectorization of  $vv^*$ , i.e.,  $u = v \otimes \bar{v}$ . This equation is of the form

$$\sum_{k=0}^N B_k z^k u = 0, \quad (15)$$



which in the literature is known as a *polynomial eigenvalue problem*. Similar to quadratic eigenvalue problems, the most common way to solve polynomial eigenvalue problems is by companion linearization, which is analyzed and generalized in [21,20]. For instance, the eigenvalues of (15) are the eigenvalues corresponding to the generalized eigenvalue problem

$$z \begin{pmatrix} I & & & \\ & \ddots & & \\ & & I & \\ & & & B_N \end{pmatrix} w = \begin{pmatrix} 0 & I & & \\ & \ddots & \ddots & \\ & & 0 & I \\ -B_0 & \cdots & -B_{N-2} & -B_{N-1} \end{pmatrix} w, \quad (16)$$

where  $w = (u^T, zu^T, z^2u^T, \dots, z^{N-1}u^T)^T$ . By selecting  $B_k$  according to the coefficients in (14), we can compute all solutions  $z, v$  to (13) by solving the eigenvalue problem (16). Other methods, such as the *Jacobi–Davidson* method could be applied directly to the polynomial eigenvalue problem without linearizing it, cf., [31].

The Hermitian rank-one matrix can be chosen similar to the choice in the previous section, i.e., the principal vector in the singular value decomposition, since the accuracy result in Lemma 4 generalizes to the polynomial case.

**Lemma 8.** Let  $(z, x)$  be an eigenpair of the polynomial eigenvalue problem (15) and  $x = v \otimes \bar{v}$ ,  $\|x\| = 1$ ,  $|z| = 1$ . Say  $\tilde{x} = x + y = u \otimes \bar{u} + q$  is an approximation of  $x$ , then the sine of the angle between the approximation  $u$  and the vector  $v$  is bounded by

$$|\sin(u, v)| \leq \sqrt{\|y\| \left( 1 + \sum_{k=0}^N \|B_k\| \right) + \|q\|}.$$

**Proof.** The proof is analogous to the proof of Lemma 4.  $\square$

**Remark 9.** For the case  $h_k = hk$ , the companion form (16) is very similar to the eigenvalue problem occurring in [6]. However, in that context it is not recognized that the eigenproblem to be solved is a polynomial eigenproblem and that the eigenvector  $u$  is the vectorization of an Hermitian rank-one matrix.

## 2.5. Notes on computation

We now clarify how we can use Theorem 3 to generate the critical surfaces, discuss computational bottlenecks and suggest possible improvements and alternatives.

Theorem 3 is an equivalence theorem between  $\varphi_1, \dots, \varphi_{m-1}$  and  $h_1, \dots, h_m$ . Hence, we can see it as a parameterization of the critical surface. In other words, if we let  $\varphi_1, \dots, \varphi_{m-1}$  run the whole interval, we will generate all critical points. In practice, we typically let the free parameter run over finite number of grid points with a grid size small enough such that we can convince ourselves of the continuity of the critical curves.

With this in mind, we state in pseudo-code how to generate the critical curves for the two-delay system.

1. FOR  $\varphi = -\pi : \Delta : \pi$
2. Find eigenpairs  $(z_k, u_k)$  of (11)
3. FOR  $k = 1 : \text{length}(z)$
4. IF  $z_k$  is on unit circle
5. Compute  $v_k$  such that  $u_k = \text{vect} v_k v_k^*$
6. Compute
7.  $\omega_k = -iv_k^* (A_2 z_k + A_0 + A_1 e^{-i\varphi}) v_k$
8. Accept critical points  $(h_1, h_2)$
9.  $h_1 = \frac{\varphi + 2p\pi}{\omega_k}, p = -p_{\max}, \dots, p_{\max}$
10.  $h_2 = \frac{-\text{Arg } z_k + 2q\pi}{\omega_k}, q = -p_{\max}, \dots, p_{\max}$
11. END
12. END
13. END

In step 1,  $\Delta$  is the stepsize of the parameter  $\varphi$ . In step 7,  $p_{\max}$  is the number of branches which should be included in the computation. Step 7 is not computationally demanding. We can select  $p_{\max}$  so large that the computation contains all relevant branches, say such that all delays smaller than some delay tolerance are found. This is possible because the delays are monotonically increasing or decreasing in the branch parameter. The generalization to more than two delays is straightforward. It involves a nesting of the outer iteration (step 1) with for-loops of the new free variables  $\varphi_k$  and computing the other delays in step 7 similar to  $h_1$ .

For DDEs of large or moderate dimension, the main computational effort in an implementation of the method is the solving of eigenvalue problems. This can be seen from the following remarks on the computational complexity.

The current general full eigenproblem solvers (for instance the QR-algorithm) require, roughly speaking, a computational effort proportional to  $N^3$  to compute the eigenvalues of an  $N \times N$ -matrix to sufficient accuracy. For [Theorem 3](#), the companion matrices are of dimension  $N = 2n^2$  and for [Theorem 7](#),  $N = 2n_m n^2$ . For both cases the computational effort is hence proportional to  $n^6$ , which is a lot for moderate or large system-dimensions  $n$ . For sparse eigensolvers the complexity is reduced, but the dimension of the system still causes a computational bottleneck.

**Remark 10.** We note that the very high computational cost for large systems, even suggests that instead of parameterizing using  $m - 1$  free variables over  $\varphi \in [-\pi, \pi]^{m-1}$ , where each evaluation costs  $n^6$ , it may be more efficient to search  $\varphi \in [-\pi, \pi]^m$  and as an additional constraint check if (or how far from) an imaginary eigenvalue is contained in the spectrum of  $A = A_0 + \sum_{k=1}^m A_k e^{-i\varphi_k}$ . Computing the spectrum of  $A$  has the computational cost  $n^3$ , which (for large  $n$ ) compensates for the additional degree of freedom which has to be scanned. We note that some form of iterative method is likely to be necessary in order to impose the additional constraint (the imaginary eigenvalue) in a reliable way.

This remark was brought to our attention by Michiel Hochstenbach.

In order to make larger problems tractable we make some initial remarks on how to adapt numerical methods for [\(11\)](#). The eigenvalues we wish to find have the following special properties we can make use of.

- (1) The eigenvalues of interest lie on the unit circle.
- (2) The matrices resulting from the companion form are sparse.
- (3) The matrices have a very special structure originating from the Kronecker construction.
- (4) Only eigenvectors of the form  $u = \text{vec } vv^*$ , i.e. a vectorization of an Hermitian rank-one matrix, are of interest.
- (5) The eigenvalues move continuously with respect to the parameters.

In this work we only exploit the first two items. The unit eigenvalue property is exploited by transforming the unit eigenvalues to the real line, i.e. we make the Cayley-transformation  $z = \frac{1+i\sigma}{1-i\sigma}$ . That is, if  $z$  is a unit eigenvalue of the eigenproblem  $Av = zBv$  then  $\sigma$  is a real eigenvalue of the eigenproblem  $(A - B)v = \sigma(iA + iB)v$ . The reason for this is that it is believed that finding real eigenvalues is an easier problem than finding eigenvalues on the unit circle. Here, we apply a method doing rational Krylov scans (cf. [\[28,29\]](#)) along the real line implemented in the command `sptarn` in the Matlab control toolbox. However, we note that further exploitation of the properties are necessary if we wish to treat large dimensional DDEs. An example of such a structure-exploitation is given in [\[17\]](#).

### 3. Examples

**Example 11** ( $n = 35, m = 3$ ). In order to determine how well the method scales with dimension we apply the method in [Section 2.5](#) to the discretization of a partial delay-differential equation. We consider the three-delay partial differential equation,

$$\begin{cases} u_t'(x, t) = u_{xx}''(x, t) + \beta(1 + \sin(3\pi x))u(x, t) - \kappa_0\delta(x - x_0)u(0, t - h_1) - \kappa_1\delta(x - x_1)u(x_1, t - h_2) \\ \quad - \kappa_2\delta(x - x_2)u(1, t - h_3) \\ u_x(0, t) = 0 \\ u_x(1, t) = 0, \end{cases} \quad (17)$$

where we pick  $\kappa_0 = \kappa_2 = 4, \kappa_1 = 10, x_0 = 1/3, x_1 = 1/2, x_2 = 3/4$  and  $\beta = 10$ . The physical interpretation of equation [\(17\)](#) is the heat equation on a rod with length one with heat production over the whole rod causing instability and three-delayed stabilizing pointwise feedbacks. This system is discretized with central difference in space with  $n$  equally distributed intervals, yielding a system of the form  $\dot{x}(t) = A_0x(t) + A_1x(t - h_1) + A_2x(t - h_2) + A_3x(t - h_3)$ , where  $x(t) \in \mathbb{R}^n$ , i.e., a three-delay DDE. For this example we pick a steplength such that the discretized system is of dimension  $n = 35$ .

The critical surface closest to the origin of the discretized system is plotted in [Fig. 3](#). The algorithm computes points on the critical surface, but for visualization purposes, we connect the points to form a surface.

The application of the algorithm in [Section 2.5](#) with the described exploitation for 390 different combinations of the parameters requires 62 min on a computer running Linux and Matlab 7 on a 2.4 GHz Intel Pentium 4 processor with 512 Mb RAM.

### 4. Conclusions

We study the conditions on the delays for multiple delay time delay systems such that the system has an imaginary eigenvalue. This is done by introducing a condition using a function  $\mathbb{L}$ , for which the imaginary eigenvalue condition has a particularly easy form. For the commensurate case the condition can be reduced to the computation of the eigenvalues of a matrix similar to that of [\[6\]](#). For the general (free) case we propose a new method, which involves solving a quadratic eigenvalue problem.

For the scalar case, the quadratic eigenvalue problem can be explicitly solved and we can find a closed explicit expression for the condition on the delays. For systems of large dimension we make initial remarks on what properties of the problem an adapted numerical scheme should exploit.

In the examples section we show the efficiency of the method by applying it to previously solved examples as well a previously unsolved three-delay example of larger dimension.

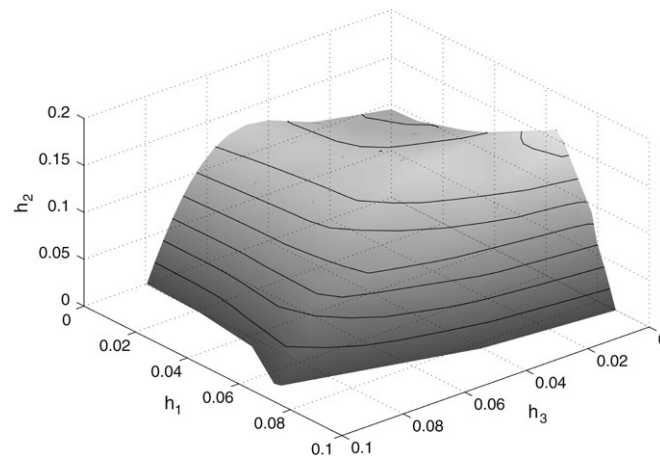


Fig. 3. Boundary of stability region for Example 11.

## Acknowledgments

We would like to thank Heike Fassbender and Tobias Damm for very valuable discussions and comments.

## References

- [1] Z. Bai, Y. Su, SOAR: A second-order Arnoldi method for the solution of the quadratic eigenvalue problem, *SIAM J. Matrix Anal. Appl.* 26 (3) (2005) 640–659.
- [2] J. Bélair, S.A. Campbell, Stability and bifurcations of equilibria in a multiple-delayed differential equation, *SIAM J. Appl. Math.* 54 (5) (1994) 1402–1424.
- [3] E. Beretta, Y. Kuang, Geometric stability switch criteria in delay differential systems with delay dependent parameters, *SIAM J. Math. Anal.* 33 (5) (2002) 1144–1165.
- [4] D. Breda, S. Maset, R. Vermiglio, Computing the characteristic roots for delay differential equations, *IMA J. Numer. Anal.* 24 (2004) 1–19.
- [5] B. Cahlon, D. Schmidt, Stability criteria for certain third-order delay differential equations, *J. Comput. Appl. Math.* 188 (2) (2006) 319–335.
- [6] J. Chen, G. Gu, C.N. Nett, A new method for computing delay margins for stability of linear delay systems, *Syst. Control Lett.* 26 (2) (1995) 107–117.
- [7] K.L. Cooke, Z. Grossman, Discrete delay, distributed delay and stability switches, *J. Math. Anal. Appl.* 86 (1982) 592–627.
- [8] A.F. Ergenc, N. Olgac, H. Fazelina, Extended Kronecker summation for cluster treatment of LTI systems with multiple delays, *SIAM J. Control Optimization* 46 (1) (2007) 143–155.
- [9] E. Fridman, Stability of systems with uncertain delays: A new “complete” Lyapunov–Krasovskii functional, *IEEE Trans. Automat. Control* 51 (5) (2006) 885–890.
- [10] P. Fu, S.-I. Niculescu, J. Chen, Stability of linear neutral time-delay systems: Exact conditions via matrix pencil solutions, *IEEE Trans. Autom. Control* 51 (6) (2006) 1063–1069.
- [11] K. Gu, V. Kharitonov, J. Chen, *Stability of time-delay systems*, in: Control Engineering, Birkhäuser, Boston, MA, 2003.
- [12] K. Gu, S.-I. Niculescu, J. Chen, On stability crossing curves for general systems with two delays, *J. Math. Anal. Appl.* 311 (1) (2005) 231–253.
- [13] J. Hale, Dynamics and delays, Delay differential equations and dynamical systems, in: Proc. Conf., Claremont/CA (USA) 1990, in: Lect. Notes Math., vol. 1475, 1991, pp. 16–30.
- [14] J. Hale, W. Huang, Global geometry of the stable regions for two delay differential equations, *J. Math. Anal. Appl.* 178 (2) (1993) 344–362.
- [15] J. Hale, S.M.V. Lunel, *Introduction to Functional Differential Equations*, Springer-Verlag, 1993.
- [16] D. Hertz, E. Jury, E. Zeheb, Simplified analytic stability test for systems with commensurate time delays, *IEE Proc., Part D* 131 (1984) 52–56.
- [17] E. Jarlebring, A quadratic eigenproblem in the analysis of a time delay system, *Proc. Appl. Math. Mech.* 6 (2006) 63–66.
- [18] C.R. Knopse, M. Roozbehani, Stability of linear systems with interval time delays excluding zero, *IEEE Trans. Autom. Control* 51 (8) (2006) 1271–1288.
- [19] J. Louisell, A matrix method for determining the imaginary axis eigenvalues of a delay system, *IEEE Trans. Autom. Control* 46 (12) (2001) 2008–2012.
- [20] S. Mackey, N. Mackey, C. Mehl, V. Mehrmann, Structured polynomial eigenvalue problems: Good vibrations from good linearizations, *SIAM J. Matrix Anal. Appl.* 28 (2006) 1029–1051.
- [21] S. Mackey, N. Mackey, C. Mehl, V. Mehrmann, Vector spaces of linearizations for matrix polynomials, *SIAM J. Matrix Anal. Appl.* 28 (2006) 971–1004.
- [22] E. Malakhovskii, L. Mirkin, On stability of second-order quasi-polynomials with a single delay, *Automatica* 42 (6) (2006) 1041–1047.
- [23] S.-I. Niculescu, *Delay Effects on Stability. A Robust Control Approach*, Springer-Verlag, London, 2001.
- [24] S.-I. Niculescu, On delay-dependent stability under model transformations of some neutral linear systems, *Int. J. Control* 74 (6) (2001) 609–617.
- [25] S.-I. Niculescu, P. Fu, J. Chen, On the stability of linear delay-differential algebraic systems: Exact conditions via matrix pencil solutions, in: Proceedings of the 45th IEEE Conference on Decision and Control, 2006.
- [26] R.D. Nussbaum, Differential-delay equations with two time lags, *Mem. Am. Math. Soc.* 205 (1978) 62.
- [27] Z. Rekasius, A stability test for systems with delays, Proc. of Joint Autom. Contr. Conf San Francisco, 1980.
- [28] A. Ruhe, Rational Krylov: A practical algorithm for large sparse nonsymmetric matrix pencils, *SIAM J. Sci. Comput.* 19 (5) (1998) 1535–1551.
- [29] Y. Saad, Variations on Arnoldi’s method for computing eigenelements of large unsymmetric matrices, *Linear Algebra Appl.* 34 (1980) 269–295.
- [30] R. Sipahi, N. Olgac, Complete stability robustness of third-order LTI multiple time-delay systems, *Automatica* 41 (8) (2005) 1413–1422.
- [31] G.L. Sleijpen, A.G. Booten, D.R. Fokkema, H.A.V. der Vorst, Jacobi–Davidson type methods for generalized eigenproblems and polynomial eigenproblems, *BIT* 36 (3) (1996) 595–633.
- [32] A. Thowsen, The Routh–Hurwitz method for stability determination of linear differential-difference systems, *Int. J. Control* 33 (1981) 991–995.
- [33] F. Tisseur, K. Meerbergen, The quadratic eigenvalue problem, *SIAM Rev.* 43 (2) (2001) 235–286.
- [34] K. Verheyden, T. Luzyanina, D. Roose, Efficient computation of characteristic roots of delay differential equations using LMS methods, *J. Comput. Appl. Math.* (2007), doi:10.1016/j.cam.2007.02.02.