2019 16th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)
Mexico City, Mexico September 11-13, 2019

An efficient algorithm for the construction of a block matrix depending on the delay Lyapunov matrix for testing stability of time-delay systems

Carlos Cuvas-Castillo
Autonomous University of Hidalgo State (UAEH), México
E-mail: carlos_cuvas@uaeh.edu.mx

Alejandro Castaño-Hernández
Master of Science student, UAEH
E-mail: ca296959@uaeh.edu.mx

Jesús Patricio Ordaz-Oliver
UAEH
E-mail: jesus_ordaz@uaeh.edu.mx

***Abstract*—This manuscript addresses a novel algorithm for the construction of a matrix, whose positivity is a necessary stability condition for linear neutral systems with commensurate delays. The condition exclusively depends on the delay Lyapunov matrix, which can be obtained by means of the so-called semianalytic method by computing a matrix exponential. The algorithm detects the repeated values of the argument of a matrix exponential (needed in the semianalytic method), reducing the computational effort. In order to show the effectiveness of the algorithm, two examples are presented.**

***Index Terms*—Linear time-delay systems, delay Lyapunov matrix, stability conditions, computational effort**

I. Introduction

The delay Lyapunov matrix has been proved its important role in linear time-delay systems in the approach of the Lyapunov-Krasovskii functionals, specifically in the case of those with a prescribed time derivative. The construction of that matrix is an important issue that has attracted the attention of the researchers. For instance, in [1], it is presented a method which establishes a set of delay-free variables, and by solving a differential equation, the delay Lyapunov matrix is obtained for systems with commensurate delays. In [2], this matrix is obtained by a piecewise linear approximation. The mentioned works, base their algorithms on the three properties of the delay Lyapunov matrix, which are the equivalent to the algebraic Lyapunov equation for the case of delay-free linear equations.

The delay Lyapunov matrix has been successfully applied for obtaining exponential estimates [1], robust stability analysis [3], Lyapunov redesign [4], optimal control [5], [6], among others.

An stability criteria based on the delay Lyapunov matrix are presented in [7], [8] and [9], for systems with multiple pointwise delays, distributed delays, and for neutral type systems, respectively. These stability conditions are determined by a block matrix which uniquely depends on the delay Lyapunov matrix, and are the equivalent of the stability criterion for delay-free systems depending on the positivity of matrix P, solution of $A^TP+PA=-W$. An interesting fact is that the stability criteria for the mentioned delay systems, have the same structure.

The contribution of this paper lies in the improvement of the computational effort in the construction of the block matrix depending on the delay Lyapunov matrix for testing the stability of neutral systems with multiple commensurate delays.

The organization of the paper is described next. In section II, some basic concepts, the properties of the delay Lyapunov matrix, stability criteria, and the semianalytic method, are introduced. In section III, the main result is presented, namely, an algorithm for the reduction of the computational effort. In section IV, two illustrative examples and additional observations are presented. The paper ends with some conclusions.

Notation: The acronym PC^1 indicates the space of piecewise continuously differentiable functions. The euclidian norm of a vector is represented by $\|\cdot\|$ and $\|\varphi\|_H=\sup_{\theta\in[-H,0]}\|\varphi(\theta)\|$. The expression $[A_{i,j}]_{i,j=1}^r$ means a square block matrix with element $A_{i,j}$ in the row i and column j, $Q>0$ represents a symmetric matrix Q, which is positive definite.

II. Preliminaries

In this work, the next class of neutral delay systems is considered,

$$\frac{d}{dt}\sum_{j=0}^{m}D_jx(t-jh)=\sum_{j=0}^{m}A_jx(t-jh),\quad t\geq 0, \tag{1}$$

where $x\in\mathbb{R}^n$, $h>0$ is the basic delay, $H=mh$ is the greatest delay, D_j and A_j, $j=0,...,m$, are constant matrices in $\mathbb{R}^{n\times n}$, with $D_0=I_{n\times n}$. The initial condition φ belongs to $PC^1\left([-H,0],\mathbb{R}^n\right)$, and the system state is $x_t:\theta\rightarrow x(t+\theta)$, $\theta\in[-H,0]$.

Definition II.1. [10] System (1) is said to be exponentially stable if there exist $\sigma>0$ and $\gamma\geq 1$ such that for every solution $x(t,\varphi)$ the following estimate holds

$$\|x(t,\varphi)\|\leq\gamma e^{-\sigma t}\|\varphi\|_H,\quad t\geq 0.$$

A. Delay Lyapunov matrix properties

The delay Lyapunov matrix $U(\cdot)$ (see for example [1], [11]) of system (1), associated with $W>0$, satisfies

- the dynamic property

$$\frac{d}{d\tau}\sum_{j=0}^{m} U(\tau - jh)D_j = \sum_{j=0}^{m} U(\tau - jh)A_j,\ \tau \geq 0, \quad (2)$$

- the symmetric property

$$U(\tau) = [U(-\tau)]^T,\ \tau \in \mathbb{R}, \quad (3)$$

- the algebraic property

$$-W = \sum_{i=0}^{m}\sum_{j=0}^{m}\left(D_i^T U((i-j)h)A_j + A_j^T U^T((i-j)h)D_i\right). \quad (4)$$

Equations (2)-(4) admit a unique solution if the Lyapunov condition is satisfied, that is, the characteristic equation of the system does not have any pair of roots whose sum is equal to zero.

B. Stability conditions

In [9], a set of necessary conditions for the stability of system (1), is presented. We remind it in the following theorem.

Theorem II.1. *[9] If system (1) is exponentially stable, then*

$$K_r(\tau_1,...,\tau_r) = [U(-\tau_i + \tau_j)]_{i,j=1}^{r} > 0, \quad (5)$$

where $\tau_k \in [0,H]$, $k = 1,...,r$, $\tau_i \neq \tau_j$ if $i \neq j$, and r is a natural number.

Remark 1. The arguments of the delay Lyapunov matrix can be chosen such that $\tau_k = \frac{k-1}{r-1}H$, $k = 1,...,r$, $r \geq 2$.

For example, for $r = 4$, K_r is given by

$$K_4\left(0, \frac{H}{3}, \frac{2H}{3}, H\right) = \begin{bmatrix} U(0) & U(\frac{H}{3}) & U(\frac{2H}{3}) & U(H) \\ U^T(\frac{H}{3}) & U(0) & U(\frac{H}{3}) & U(\frac{2H}{3}) \\ U^T(\frac{2H}{3}) & U^T(\frac{H}{3}) & U(0) & U(\frac{H}{3}) \\ U^T(H) & U^T(\frac{2H}{3}) & U^T(\frac{H}{3}) & U(0) \end{bmatrix}.$$

Remark 2. It is important to mention that Theorem II.1 is the starting point of the necessary and sufficient stability conditions for equations with discrete [7] and distributed delays [8].

C. Semianalytic method

The semianalytic method [1] is applicable when the delay-system has commensurate delays, which is the case of (1).

For $\xi \in [0,h]$, the next set of $2m$ auxiliary variables is defined:

$$X_j(\xi) := U(\xi + jh), \quad j = -m,...,0,...m-1. \quad (6)$$

These matrices satisfy equations (2)-(4). Thus, we have

$$\begin{aligned} \sum_{i=0}^{m} X'_{j-i}(\xi)D_i &= \sum_{i=0}^{m} X_{j-i}(\xi)A_i, \quad j = 0,...,m-1, \\ \sum_{i=0}^{m} D_i^T X'_{j+i}(\xi) &= -\sum_{i=0}^{m} A_i^T X_{j+i}(\xi), \quad j = -m,...,-1, \end{aligned} \quad (7)$$

and

$$\begin{aligned} X_{j+1}(0) &= X_j(h), \quad j = -m,...,-1,0,1,...,m-2, \\ -W &= \sum_{j=0}^{m}\left(D_j^T X_0(0)A_j + A_j^T X_0(0)D_j\right) \\ &+ \sum_{j=0}^{m-1}\sum_{i=j+1}^{m}\left(D_i^T X_{i-j-1}(h)A_j + D_j^T X_{i-j-1}^T(h)A_i\right. \\ &\left.+A_j^T X_{i-j-1}^T(h)D_i + A_i^T X_{i-j-1}(h)D_j\right). \end{aligned} \quad (8)$$

One can notice that (7) is a set of differential equations with boundary conditions (8). After vectorization [12], (7) and (8) are expressed as

$$X'(\xi) = LX(\xi), \quad \xi \in [0,h], \quad (9)$$

and

$$MX(0) + NX(h) = W_v,$$

respectively, where $X(\cdot)$ is the extended vector of variables (6), i.e.

$$X(\xi) := \left(\text{vec}(X_{-m}(\xi))^T \quad \dots \quad \text{vec}(X_{m-1}(\xi))^T\right)^T,$$

and $L \in \mathbb{R}^{2mn^2 \times 2mn^2}$, M, N and W_v are the result of the rearrangement of (7) and (8).

The solution of (9) is expressed as

$$X(\xi) = e^{L\xi}X(0), \quad \xi \in [0,h] \quad (10)$$

with initial condition

$$X(0) = \left(M + Ne^{Lh}\right)^{-1} W_v.$$

Next, the computation of the delay Lyapunov matrix is established.

Lemma II.2. *Consider the differential equations (7), with boundary conditions (8). If there exists a unique set of solutions (6), then, there exists a unique delay Lyapunov matrix, associated with W, defined on $[-H,H]$ by the expressions*

$$U(\xi + jh) = X_j(\xi), \quad \xi \in [0,h], \quad j = -m,...,0,...m-1.$$

III. Main result

After the introduction of basic concepts, here it is presented the main result of the paper, an algorithm for the reduction of the computational cost in the construction of matrix K_r.

A. Algorithm

In particular, in [13], [14], the values of τ_k, $k = 1,...,r$, are taken according to Remark 1. Hence, constructing matrix K_r requires the delay Lyapunov matrix evaluated at r pointwise values. For instance, if $r = 4$, K_r requires the delay Lyapunov matrix $U(\tau)$ evaluated at $\tau_1 = 0$, $\tau_2 = H/3$, $\tau_3 = 2H/3$ and $\tau_4 = H$.

However, notice that some of these values cannot be directly evaluated in equation (10), since the matrix exponential admits only arguments on the interval $[0,h]$. Therefore, a valid argument, ξ, and j, must be calculated in order to obtain $U(\tau)$ from variable X_j in (6).

The matrix $U(\tau_k)$, with τ_k as in Remark 1, can be computed through the next algorithm:

- Obtain the variable X_j, and the value ξ_k, based on Lemma II.2, this is

$$j = \lfloor \tau_k / h \rfloor, \tag{11}$$

$$\xi_k = \tau_k - jh. \tag{12}$$

- Evaluate expression (10) with ξ_k and obtain $U(\tau_k)$ from $X_j(\tau)$.
- For the last value of τ_k, $\tau_r = H$, it is necessary to establish $\xi_r = h$, and $j = m - 1$.

It is worthy of mention that, in this procedure, it is possible to obtain the same value of ξ, for different values of j, which implies the computation of the matrix exponential in (10) repeated times. This, of course, derives in unnecessary computational effort.

The main result of the work is the improvement, in computational effort terms, of the construction of matrix K_r for systems of the form (1) by employing the semianalytic method.

First, from (12), notice that

$$\xi_k = \left(\frac{(k-1)m}{r-1} - \left\lfloor \frac{(k-1)m}{r-1} \right\rfloor \right) h. \tag{13}$$

The first term on the right hand side is rewritten as

$$\frac{(k-1)m}{r-1} = \left\lfloor \frac{(k-1)m}{(r-1)} \right\rfloor + \frac{mod \left[\frac{(k-1)m}{(r-1)} \right]}{r-1}.$$

Substituting this equation into (13) yields to

$$\xi_k = mod \left[\frac{(k-1)m}{r-1} \right] \frac{h}{r-1}.$$

Based on this equation, it is evident that, in some cases, repeated values of ξ_k can be calculated.

The presented algorithm avoids the repeated computation of these values (if there exist repeated values) in (10), hence, an improvement in time computation of K_r is expected.

Compute the quantity

$$p1 = gcd(m, (r-1)).$$

The number of different values of ξ_k, excluding $\xi_r = h$, is given by

$$p2 = \frac{r-1}{p1}.$$

Actually, each one of these values are repeated $p1$ times. These data are important in the presented algorithm in order to reduce the computational effort. The delay which is the base of this algorithm is given by

$$h_b = p1 \frac{h}{r-1}.$$

The value $p1$ is important in the calculation of h_b and in the construction of an auxiliary array of ξ_k.

Since h_b is a basic delay, the computation of the matrix exponential in (10) is reduced as $e^{Lh_b i} = \left(e^{Lh_b}\right)^i$, $i = 1, \ldots, p2 - 1$.

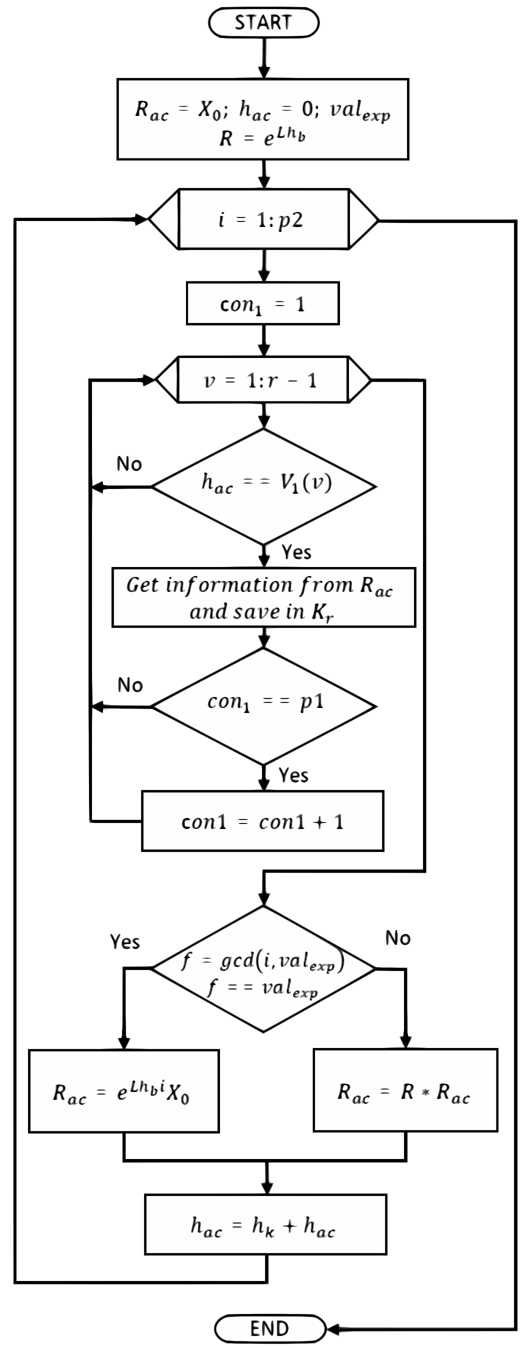

Fig. 1. Algorithm flow chart

Hence, the knowledge of h_b, added to the computation of variable X_j, reduces the computation of repeated values to just one.

Figure 1 shows the flow chart of the proposed algorithm.

For example, suppose that $m = 9$ and $r = 13$, then $p1 = 3$, $p2 = 4$ and $h_b = 3h/12$. The array of ξ_k is $V_1 h/12$ with

$$V_1 = \begin{bmatrix} 0 & 9 & 6 & 3 & 0 & 9 & 6 & 3 & 0 & 9 & 6 & 3 & 12 \end{bmatrix}.$$

The last value of this vector is $\xi_{13} = 12$, and $j = 8$. Notice that $\xi_1 = 0$ in every case.

The algorithm first detects the $p1$ positions which contain the values of $\xi_k = 0$, and using these positions, calculates the values of j, with (11), in order to obtain $X_j(0)$, which are introduced appropriately in K_{13}.
Right away, with the computation of $X(h_b)$ using (10), the algorithm again detects the $p1$ positions which contain the values h_b in array of ξ_k, trough the localization of value $p1$ in vector V_1.
Next, it is calculated $X(2h_b)$, with $e^{2Lh_b} = (e^{Lh_b})^2$, and the process of localization in vector V_1 of values $2p1$, is carried out again. These steps are repeated $p2$ times.
In order to avoid numerical errors in the computation of e^{iLh_b}, we have proposed that for every val_{exp} times, the matrix exponential should be computed again.

B. Additional Observations

The presented algorithm can be used for linear systems with multiple neutral and distributed delays, where the semianalytic method can be applied as well. That case is not studied in this paper due to the auxiliary delay-free variables depend directly on the form of the kernel of the distributed term, and many cases can be addressed. The presented algorithm has more impact on the obtention of stability maps where the parameters are the delays of the systems, as in the examples (section IV); but it can be employed in methods where the delay Lyapunov matrix has to be evaluated in multiple discrete points. Although the values of r in the examples of the next section, are relatively large, Theorem II.1 and Remark 1 establish the evaluation of condition (5) for values $r \geq 2$, therefore, these values are justified.

IV. Illustrative Examples

In this section we present two illustrative examples that show the performance of the developed algorithm. In each example, 120×120 points of the space of parameters are evaluated, besides $val_{exp} = 5$. The points in the figures indicate the values of the pairs (h_1, h_2) where the conditions $K_r(\tau_1, ..., \tau_r) > 0$, $\tau_j = \frac{j-1}{r-1}H$, $j = 1, ..., r$, are satisfied. The entire procedure was implemented in Matlab 2016, in a Lenovo Y530, intel core i7. For each example and for every value of r, the computer was only executing Matlab, looking for similar operation conditions, such that a fair comparison in terms of execution time, was done. In the tables, t_{normal} and $t_{algorithm}$ mean elapsed time without and with the algorithm proposed here, respectively.

Example IV.1. We take the example exposed in [15] given by

$$\dot{x}(t) = -1.3x(t) - x(t-h_1) - \frac{1}{2}x(t-h_2),$$

where $h_1 \geq 0$ and $h_2 \geq 0$ are the parameters, the objective is the obtainment of the stability map (h_1, h_2). In [14], the condition K_r is applied for $r = 8$. Here, we take the values $r = 100, 200, 300$.

First, we recall the algorithm from [14], [13], which minimizes the dimension of the auxiliary system (6)

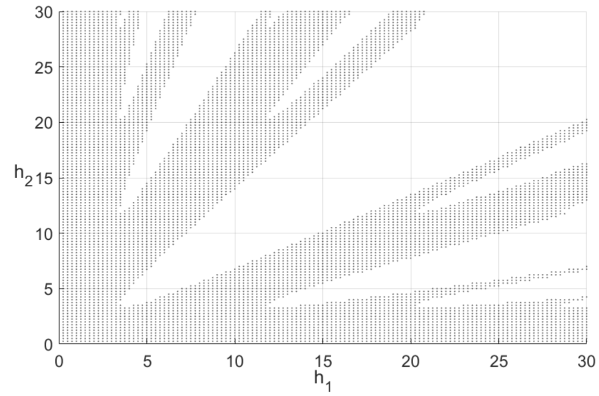

Fig. 2. Region where stability conditions $K_r > 0$ are satisfied, for $r = 100, 200, 300$, Example IV.1

- Divide each axis in $N+1$ points with separation Δ.
- Establish a counter for every axis, i.e. $h_1^{c_1} = c_1\Delta$ and $h_2^{c_2} = c_2\Delta$, $c1_, c2 = 0, ..., N$. The basic delay is taken as $h = \gcd(c_1, c_2)\Delta$.
- Compute the value $m = \max(c_1, c_2)/\gcd(c_1, c_2)$, after vectorization, the dimension of system (7) is $2mn^2$.
- Apply Lema II.2, considering that the values of matrices for delays different to $h_1^{c_1}$ and $h_2^{c_2}$, are equal to zero.

In this case, $N = 120$. The results in the stability map are the same for every value of r, and depicted in figure 2. The results are shown in Table I.

TABLE I
COMPUTATIONAL TIME COMPARISON

r	t_{normal} (min)	$t_{algorithm}$ (min)
100	33.52	5.93
200	65.8	14.7
300	96.5	19.01

Example IV.2. The second example is the neutral system with three delays [13] of the form

$$\dot{x}(t) + 0.8\dot{x}(t-h_1) + 0.15\dot{x}(t-h_2) = x(t) - 1.5x(t-h_1) \\ +2x(t-h_2) - 5x(t-(h_1+h_2)),$$

where $h_1 \geq 0$ and $h_2 \geq 0$ are the parameters. The matrix $U(\cdot)$ was computed using the main ideas of the algorithm presented in the first example. Again, $r = 100, 200, 300$.

The results for every value of r, are the same (figure 3). Table IV.2 shows the elapsed time comparison for each case.

TABLE II
COMPUTATIONAL TIME COMPARISON

r	t_{normal} (min)	$t_{algorithm}$ (min)
100	75.4	13.1
200	147.49	31.3
300	227.42	42.75

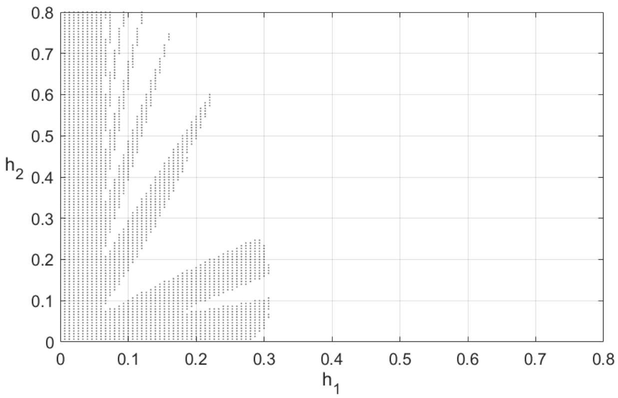

Fig. 3. Region where stability conditions $K_r > 0$ are satisfied, for $r = 100, 200, 300$, Example IV.2

V. CONCLUSIONS

This paper reports the development and implementation of an algorithm that reduces the computational effort required for the obtainment of the delay Lyapunov matrix employing the semianalytic method. The construction of K_r requires of the matrix $U(\cdot)$ for stability analysis of linear time-delay equations. The algorithm avoids the repeated computation of a matrix exponential, identifying a basic delay for the computation of that matrix. Therefore, computational time reduces considerably, the examples give evidence of the algorithm performance.

ACKNOWLEDGMENTS

This work was supported by PRODEP-México UAEH-PTC-785, 511-618-8661.

REFERENCES

[1] V. Kharitonov, *Time-delay systems: Lyapunov functionals and matrices*. Springer Science & Business Media, 2012.

[2] G. Ochoa, S. Mondie, and V. Kharitonov, "Time delay systems with distributed delays: critical values," *IFAC Proceedings Volumes*, vol. 42, no. 14, pp. 272–277, 2009.

[3] A. V. Egorov and S. Mondié, "The delay lyapunov matrix in robust stability analysis of time-delay systems," *IFAC-PapersOnLine*, vol. 48, no. 12, pp. 245–250, 2015.

[4] L. Rodríguez-Guerrero, S. Mondié, and O. Santos-Sánchez, "Guaranteed cost control using lyapunov redesign for uncertain linear time delay systems," *IFAC-PapersOnLine*, vol. 48, no. 12, pp. 392–397, 2015.

[5] O. Santos and S. Mondie, "On the optimal control of time delay systems: a complete type functionals approach," in *2007 46th IEEE Conference on Decision and Control*. IEEE, 2007, pp. 6035–6040.

[6] O. Santos and S. Mondié, "Guaranteed cost control of linear systems with distributed delays: A complete type functionals approach," *International Journal of Control, Automation and Systems*, vol. 8, no. 3, pp. 497–505, 2010.

[7] A. V. Egorov, "A new necessary and sufficient stability condition for linear time-delay systems," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 11 018–11 023, 2014.

[8] A. V. Egorov, C. Cuvas, and S. Mondié, "Necessary and sufficient stability conditions for linear systems with pointwise and distributed delays," *Automatica*, vol. 80, pp. 218–224, 2017.

[9] M. A. Gomez, A. V. Egorov, and S. Mondié, "Necessary stability conditions for neutral-type systems with multiple commensurate delays," *International Journal of Control*, pp. 1–12, 2017.

[10] R. E. Bellman and K. L. Cooke, *Differential-difference equations*. Rand Corporation, 1963.

[11] G. Ochoa, J. Veláquez-Veláquez, V. Kharitonov, and S. Mondié, "Lyapunov matrices for neutral type time delay systems," *IFAC Proceedings Volumes*, vol. 40, no. 23, pp. 244–249, 2007.

[12] P. Lancaster and M. Tismenetsky, *The theory of matrices: with applications*. Elsevier, 1985.

[13] M. A. Gomez, C. Cuvas, S. Mondié, and A. V. Egorov, "Scanning the space of parameters for stability regions of neutral type delay systems: A lyapunov matrix approach," in *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 2016, pp. 3149–3154.

[14] C. Cuvas, A. Ramírez, L. Juárez, and S. Mondié, "Scanning the space of parameters for stability regions of time-delay systems: A lyapunov matrix approach," in *Delays and Interconnections: Methodology, Algorithms and Applications*. Springer, 2017.

[15] R. Sipahi, S.-I. Niculescu, C. T. Abdallah, W. Michiels, and K. Gu, "Stability and stabilization of systems with time delay," *IEEE Control Systems Magazine*, vol. 31, no. 1, pp. 38–65, 2011.