Journal of the Franklin Institute 361 (2024) 106643

Contents lists available at ScienceDirect

Journal of the Franklin Institute

journal homepage: www.elsevier.com/locate/fi

On the μ-analysis and synthesis for uncertain time-delay systems with Padé approximations☆

Rafael Fernandes Pinheiro [a,*], Diego Colón [b]

[a] Center for Innovative Care and Health Technology (ciTechCare), Polytechnic of Leiria, Rua de Santo André, 66-68, Leiria, 2410-541, Portugal
[b] Laboratory of Control and Automation, University of São Paulo, Av. Prof. Luciano Gualberto, Travessa 3, No. 158, São Paulo, 05508-900, Brazil

ARTICLE INFO

Keywords:
Robust control
Structured singular value
μ-analysis and synthesis
Time-delay systems
Uncertain TDS

ABSTRACT

Real problems in control engineering usually involve many uncertainties and delays. In the search for solutions which deal more adequately with these problems, this paper presents contributions to the theory of the μ-analysis and synthesis applied to uncertain linear fixed time time-delay systems by using Padé approximations. A new necessary and sufficient condition for robust stability for a class of uncertain time-delay systems is presented. From this condition, a novel robust controller synthesis technique is obtained. Furthermore, contributions are presented on the convergence theory of Padé approximations applied to the μ-theory via parametrized optimization technique. In order to better understand the theory presented and verify the effectiveness of the results, examples and comparisons are proposed. Finally, in the conclusions, new lines of research and application are pointed out.

1. Introduction

Time-delay systems (TDS) are present in a multitude of physical phenomena, for which reason they have been widely studied in engineering [1–6] and science [7–9]. The time delay may be time-variant or even uncertain [10–14] which may fall in a category studied by robust control theory. For the control of nonlinear systems with uncertain time delays, important results have appeared in the literature [15–17]. In addition, advanced approaches to robust adaptive control of uncertain nonlinear systems with delay have been presented [18]. Many techniques for robust stability analysis and controller design have been developed over the years, where one of them is structured singular value (SSV) theory, also known as μ-analysis and synthesis, both developed by John Doyle [19]. According to Skogestad [20], μ-analysis is a technique that can provide necessary and sufficient conditions for robustness of stability and performance of linear time-invariant systems (LTI). In computational terms, there are difficulties in working with TDS, since they are infinite dimensional systems represented by functional differential equations [4]. In order to circumvent this difficulty, approximations for the time delay are used, such as: Padé approximations [21], Taylor series [22], among others [23–25]. In this paper, one presents original techniques of analysis and feedback control design for uncertain linear time-invariant TDS systems (ULTITDS) based on μ-analysis and synthesis, which were revisited by the authors in an attempt to obtain less conservative results than the ones found in literature. This work presents the way to proofs of conjectures raised in the papers [26,27] by the same authors.

☆ This work was funded by Portuguese national funds, Portugal provided by Fundação para a Ciência e Tecnologia (FCT-UIDB/05704/2020) and in the scope of the research project 2 ARTs - Acessing Autonomic Control in Cardiac Rehabilitation (PTDC/EMD-EMD/6588/2020) co-financed by the Portuguese Foundation for Science and Technology (FCT), Portugal.

* Corresponding author.
E-mail addresses: rafael.f.pinheiro@ipleiria.pt (R.F. Pinheiro), diego@lac.usp.br (D. Colón).

https://doi.org/10.1016/j.jfranklin.2024.01.044
Received 7 May 2023; Received in revised form 16 November 2023; Accepted 24 January 2024
Available online 26 January 2024

The concept of SSV provides necessary and sufficient conditions for robustness of stability and performance for finite order uncertain linear time-invariant systems (ULTI). There are several numerical toolboxes, as for example, the Robust Control Toolbox of MATLAB® by Mathworks®, which implement well consolidated algorithms for finding the SSV, as well as designing with it. The methods presented here allow for the use of such numerical toolboxes with ULTITDS, which is a major advantage to the practitioner. Despite the fact that robust control theory has already been applied to TDS systems [28–30], the approach presented here adds another gun to the arsenal for attacking problems in control system design of TDS uncertain systems.

1.1. Related works

Wang and Skogestad [31] use μ-analysis and synthesis with Smith predictor for ULTITDS by approximating the delay term by Padé approximations (in the example presented, a fourth order approximation is used). Despite being the simplest method found, one does not have any indication that the method would provide robustness of stability for a higher order Padé approximation. Probably a higher conservatism is necessary in order to achieve robustness. The use of Padé to approximate the time-delay term mixed with the μ-analysis and synthesis is therefore not new [28,31–33].

Huang and Zhou [11] take a different approach via μ-analysis by using a constant time-delay term $e^{-j\tau\omega}$ as an uncertainty by including the phase information in the Δ uncertainty matrix. The authors claim that this method provide less conservative results than other works cited therein. Despite of guaranteeing robustness of stability, treating the time-delay term as an uncertain transfer function introduce conservatism in the design process.

Our approach for the problem, differently from the references above cited, is to use the SSV tool in analysis and synthesis of ULTITDS by analysing a truncated sequence of functions in the frequency domain. This sequence is obtained by substituting the delay term $e^{-j\omega\tau}$ by its Padé approximations $[L_k, M_k]$ and taking an upper bound $m_k(j\omega)$ for the SSV of those approximations. Neither the SSV function nor the upper bound function $m_k(j\omega)$ have closed form, but those ones can be efficiently numerically calculated, as they conduct to convex optimization problems. Several numerical toolboxes can calculate the functions $m_k(j\omega)$, as for example the Robust Control Toolbox of Mathworks' MATLAB®. This method was applied in some classes of Lurie type problems with time-delay, which can be reduced to ULTITDS [26,34] and the results were less conservative if compared with others in the literature.

1.2. Contributions

In order to better clarify the contributions of this paper, they have been listed in topics, as follows:

- Theorem 4 is the main result in the paper, which is a generalization of Theorem 3 [20] for ULTITDS (Theorem 3 is only valid for systems with no time-delay). The extension consists in approaching ULTITDS, as said above, by replacing the time delay term $e^{j\omega\tau}$ by a sequence of its Padé approximations. This theorem enables the proof of Theorem 9 which is the second main result of the paper, and deals with the synthesis of closed-loop controllers. Theorems 4 and 9, in a version for non-linear systems, were treated as unproved conjectures in [26,35] and were used as such to analyse stability and design feedback controllers for a class of Lurie type systems [36]. Thus, the proof of Theorem 4 can be adapted to the proof of those conjectures.
- In order to prove Theorems 4 and 9, several other theorems and lemmas are necessary, some of them which are proved in the present paper. Lemmas 1–4 assert the convergence and uniform convergence of some sequences of matrices functions necessary for the main theorems. Despite of being simple applications of classical theorems of Complex Analysis and Matrix Analysis, they are not present in the literature, as far as the authors know.
- Theorem 5 proves that the upper bound for the SSV function $\mu(\omega)$ as used by Robust Control Toolbox of MATLAB®, and which is called $m(j\omega)$ in the present paper, is a continuous function. Although it is known that $\mu(\omega)$ is continuous in some cases, there is nothing about $m(j\omega)$, as far as the authors know. It is an interesting result that can be useful for the Control community. Even if it was an already published (and not well known) result, one is offering here another proof by using the Berge's theorem. This last theorem is largely unknown in the Control community (better known in Economics community [37]). Theorem 6 is proved in a similar way, but is more adequate to the proofs of Theorems 4 and 9.
- Theorem 9 is also a contribution of this paper. Theorem 9 is an extension of Theorem 4 and is related to the μ design tool known as DK-iteration.

Thus, all the theorems, lemmas and algorithms presented in the paper, with exception of Theorems 1, 2, 3 and 8 are original contributions.

1.3. Paper organization

The rest of the paper is structured as follows: in Section 2, the problem statement is made in detail. The basic concepts for the construction of the results are explained in Section 3. Section 4 presents the results for the robustness analysis for ULTITDS and examples. Section 5 presents the results for μ-synthesis for ULTITDS, as well as examples. Finally, in Section 6, the conclusions are presented, including a lists of the main findings and suggestions of future investigations and applications based in the results of this paper.

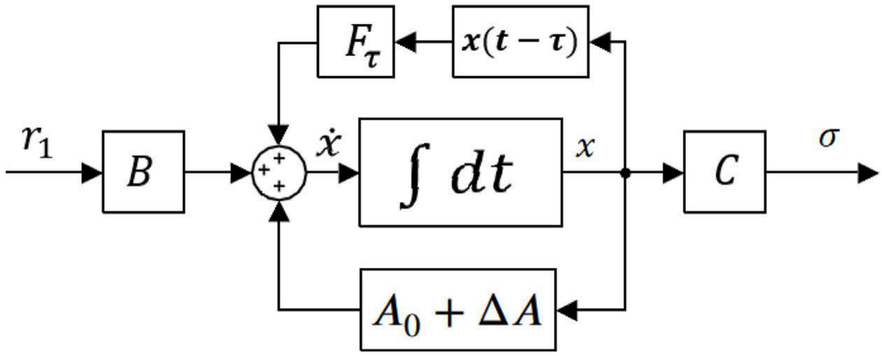

Fig. 1. Linear time-invariant TDS.

2. Problem statement

Uncertain linear time-invariant time-delay systems (ULTITDS) are treated in this paper. It is adopted for the rest of the work the state-space representation in Eq. (1):

$$\begin{cases} \dot{x} = (A_0 + \Delta A)x + F_\tau x(t-\tau) + Br_1, \\ \sigma = Cx, \end{cases} \tag{1}$$

where $x \in \mathbb{R}^n$ is the state vector, $A_0, A_1, \dots, A_m \in \mathbb{R}^{n\times n}$ are constant matrices, $\Delta A = \sum_{j=1}^m A_j\delta_j$ represents parametric uncertainties using normalized uncertain parameters δ_j such that $|\delta_j| \le 1$ for $j = 1, \dots, m$. One assumes that $\tau > 0$ is a real constant delay. The vector $r_1 \in \mathbb{R}^m$ can be an input signal and the matrices $F_\tau \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{m\times n}$ are also constant. The initial condition is defined by:

$$x(\theta) = \phi(\theta), \quad \theta \in [-\tau, 0], \quad \phi \in C^\infty[-\tau, 0]. \tag{2}$$

which is a smooth function in the interval $[-\tau, 0]$, which is a typical initial condition for functional differential equations [4,8].

There are many practical problems that can be put into the form of the Eq. (1), for example: applications in helicopter stabilization systems considering transmission delay [14]; systems involving mass–spring–damping if uncertainties and transmission delays are taken into account [38,39]; and mathematical models of neurological diseases involving memory loss [40,41]. This last example will be explored in more detail in Section 5 (see Example 5). Other examples can also be found in the classical Ref. [8]. A block diagram of the system in Eq. (1) is shown in Fig. 1.

The problem at hand is to find a generalization of the concept of structured singular value (SSV), which is represented by the real function $\mu(j\omega)$, that could be used for the class of ULTITDS in analysis and synthesis problems. Also known as μ-analysis, the function $\mu(j\omega)$ is a practical way to check if an ULTI (that is, a uncertain linear time-invariant system) is robustly stable, that could be an open or closed loop system. It is only necessary to check if $\mu(j\omega) < 1$ in order to guarantee robustness of stability. It is also possible to use this tool to verify robustness of performance [20]. One also wants to use this generalization of the μ function, that one calls $\mu_{deltay}(j\omega)$, for designing robust closed-loop controllers for ULTITDS, which is known as μ-synthesis. Let us also suppose that the controllability of the pairs $(A_0 + \Delta A, B)$ and the observability of the pairs $(C, A_0 + \Delta A)$ are guaranteed.

Unfortunately, the function $\mu(j\omega)$ never has a closed form, and can only be determined numerically. Worse than that, this numerical calculation is frequently not practical, as it involves an optimization problem that is not convex [20]. Yet in the first years of the proposition of this concept (that is, the structured singular value) this problem was solved by finding a superior limit function $m(j\omega)$ that can be efficiently numerically calculated, as it is a convex optimization problem. If one guarantees that $m(j\omega) < 1$, a consequence is that $\mu(j\omega) < 1$, that is, the robustness is guaranteed. This upper bound is then easily calculated numerically and is implemented in practical toolboxes, like the Robust Control Toolbox of MATLAB® by Mathworks. It will be shown that it is possible to define $m_{delay}(j\omega)$ and $\mu_{delay}(j\omega)$ for ULTITDS by a limit process that have the same relation. In the next section, one presents the basic concepts necessary to prove the results of the paper.

3. Basic concepts

3.1. Parametrized optimization

A classical constrained maximization problem is given by:

Find $x^* = argmax_{x\in X} f(x)$,

subject to $g_i(x) \le 0$, for $i = 1, \dots, m$,

Table 1
Padé table for e^s.

$M \setminus L$	0	1	2
0	$\frac{1}{1}$	$\frac{1+s}{1}$	$\frac{2+2s+s^2}{2}$
1	$\frac{1}{1-s}$	$\frac{2+s}{2-s}$	$\frac{6+4s+s^2}{6-2s}$
2	$\frac{2}{2-2s+s^2}$	$\frac{6+2s}{6-4s+s^2}$	$\frac{12+6s+s^2}{12-6s+s^2}$

where X is the *decision space*, $f : X \to \mathbb{R}$ is the *cost function* and $g_i : X \to \mathbb{R}$, for $i = 1, \ldots, m$ are the *constraint functions*. Those are standard definitions that can be found in many Refs. [42,43]. Consider now a class of optimization problems that depend on a set of parameters $\theta_1, \theta_2, \ldots, \theta_p$, that can be grouped in a vector $\theta = [\theta_1, \ldots, \theta_p]^T$. Let us define a *parameter space* $\Theta \subset \mathbb{R}^p$ where θ belongs. A *parametrized optimization problem* is defined by [37] (page 497):

Find $x^*(\theta) = \text{argmax}_{x \in X} f(x, \theta)$,

subject to $g_i(x, \theta) \leq 0$, for $i = 1, \ldots, m$,

where the cost function is of the form $f : X \times \Theta \to \mathbb{R}$ and the constraint functions are $g_i : X \times \Theta \to \mathbb{R}$, for $i = 1, \ldots, m$. Note that the parameters in θ are not *decision variables*, that is, one is not interested in optimizing in them. Finished the optimization problem, two functions of θ will be delivered: (1) the function $x^* : \Theta \to X$ represented by $x^*(\theta)$, that gives, for each parameter vector θ, an optimum point and (2) the function $v(\theta) = \sup_{x \in X} f(x, \theta)$ that gives the maximum value corresponding to $x^*(\theta)$. In principle, nothing can be said about the smoothness properties of those functions. Many theorems about those properties can be found [37]. In particular, one is interested in the following theorem:

Theorem 1 (*Berge's Continuous Maximum Theorem, [37]*)**.** *Consider the general constrained maximization problem:*

$$\max_{x \in G(\theta)} f(x, \theta).$$

If the objective function $f : X \times \Theta \to \mathbb{R}$ is continuous and the set function $G : \Theta \to 2^X$ is continuous and compact-valued, then the value function $v : \Theta \to \mathbb{R}$ such that $v(\theta) = \sup_{x \in G(\theta)} f(x, \theta)$ is continuous and the set function of optimal values, that is $x^*(\theta) = \text{argmax}_{x \in G(\theta)} f(x, \theta)$ is nonempty, compact-valued and upper semi-continuous.

Proof. It can be found in [44]. $\square$

One point of clarification is needed here: the function $G : \Theta \to 2^X$ is a set function, that is, it maps, for each value of θ, a subset of X. The set 2^X is known as the *power set* of X and is the set of all the subsets of X, including the empty set [37]. The $G(\theta)$ represents a subset of X. The Berge's theorem asks for G continuous and all the subsets $G(\theta)$ being compact. In the present paper, the parameter will be the scalar $\theta = \omega$, that is, the angular frequency in the frequency response of a system and are going to be used in the proofs of Theorems 5 and 6.

3.2. Padé approximation

Padé approximation is a technique for approximating general functions using rational functions [45,46]. The technique was developed by Henri Padé around 1890 and improved by others. In the following, it is presented some results that will be useful for the theorems' proofs. Given an analytic function on a region of $\mathbb{C}$ given by a power series:

$$f(s) = \sum_{i=0}^{\infty} c_i s^i, \tag{3}$$

a Padé approximation of order $[L, M]$ is a rational function given by:

$$[L/M] = \frac{a_0 + a_1 s + \cdots + a_L s^L}{b_0 + b_1 s + \cdots + b_M s^M}, \tag{4}$$

where L is the degree of the polynomial in the numerator and M is the degree of the polynomial in the denominator. In the case of the exponential function e^s, Table 1 presents some Padé approximations for different values of L and M [21]. In particular, one is interested in the approximations with $L = M$, that is typically used in applications in control problems. Regarding the convergence of the Padé approximation, Definition 1 and Theorem 2 are essential to the proofs of our results:

Definition 1. According to Freitag and Busam [47], page 106, let $S \subset \mathbb{C}$ be a non-empty subset in the complex plane. Given a sequence of complex-valued functions $\{f_k(s)\}_{k \in \mathbb{N}}$, the sequence is said to be *uniformly convergent* or *to converge uniformly* over S if there exists a function f defined in S such that for every $\epsilon > 0$ independent of s, there exists a positive integer N also independent of s such that for all $k \geq N$ and $s \in S$, one has:

$$|f_k(s) - f(s)| < \epsilon.$$

In section 6.5 of [21], several theorems related to convergence of Padé approximations are presented, which are valid except in a set of zero measure. In particular, for this paper one uses an approximation with equal degree in the numerator and denominator, such that the more adequate theorem is the following:

Theorem 2. *Let $f(s)$ be a meromorphic function. Suppose that ϵ and δ are given positive numbers. Then there exists a positive integer M_0 such that any $[M/M]$ Padé approximation satisfies:*

$$\|f(s) - [M/M]\| < \epsilon,$$

for all $M > M_0$ on any compact set of s-plane except for a set $\mathcal{E}_M$ of measure less than δ.

Proof. See [21], page 269. □

The reader should note that this convergence is uniform, as δ and ϵ do not depend on s. Also, the measure of the set $\mathcal{E}_M < \delta$ is arbitrarily small [21] (see page 269). One can also apply Theorem 6.4.5 of the same reference (see page 262). Regarding the function e^{-Ts}, the paper [46] presents several stronger convergence results, including error bounds. On the other hand, for the present paper, Theorem 2 will suffice. Results in the location of the poles and zeros of Padé approximations are presented in [48,49] that can be used for determine the nominal stability of ULTITDS.

3.3. Linear fractional transformation and uncertainty in linear systems

A robust control system with normalized uncertainties can always be put in the form of Fig. 11, where K is a closed-loop controller and $\tilde{\Delta}$ is the matrix of uncertainties, in which $\|\tilde{\Delta}\|_\infty \leq 1$ [20]. The signals are: w is a vector of external inputs, z is a vector of performance output, p and q are virtual signals that connect uncertainties and the plant, v is the vector of controller inputs and r_1 is the vector of control signals (generated by the controller).

In order to calculate the closed-loop system that include controller and uncertainties, the concepts of upper and lower linear fractional transformations (LFT) are important. Consider a matrix of transfer functions $H(s)$ of dimensions $(n_1 + n_2) \times (m_1 + m_2)$ subdivided as follows:

$$H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix},$$

and $K_1 \in \mathbb{C}^{m_1 \times n_1}$ and $K_2 \in \mathbb{C}^{m_2 \times n_2}$. The formulas for lower and upper LFT are [20]:

$$F_L(H, K_1) \stackrel{\Delta}{=} H_{11} + H_{12}K_1(I - H_{22}K_1)^{-1}H_{21}, \tag{5}$$

$$F_U(H, K_2) \stackrel{\Delta}{=} H_{22} + H_{21}K_2(I - H_{11}K_2)^{-1}H_{12}. \tag{6}$$

The control synthesis problem consists in finding the controller $K(s)$ in order to guarantee robustness of stability and robustness of performance. The information of uncertainties in the plant (before normalization), the performance specification and the bounds in the control signal are contained in the matrix $P(s)$, which is known as *augmented plant* [20]. In this paper, the class of plants that will be treated is given in state-space form (the time delay will be inserted afterwards) given by:

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \sigma \end{bmatrix} &= \begin{bmatrix} A_0 + \sum_{j=1}^m \delta_j A_j & B_0 + \sum_{j=1}^m \delta_j B_j \\ C_0 + \sum_{j=1}^m \delta_j C_j & D_0 + \sum_{j=1}^m \delta_j D_j \end{bmatrix} \begin{bmatrix} x \\ r_1 \end{bmatrix} \\ &= \left(\begin{bmatrix} A_0 & B_0 \\ C_0 & D_0 \end{bmatrix} + \sum_{j=1}^m \delta_j \begin{bmatrix} A_j & B_j \\ C_j & D_j \end{bmatrix} \right) \begin{bmatrix} x \\ r_1 \end{bmatrix}. \end{aligned} \tag{7}$$

where A_0, B_0, C_0 and D_0 are the nominal matrices, and A_j, B_j, C_j and D_j for $j = 1, \ldots, m$ are coefficient matrices with appropriate dimensions for the real-valued normalized uncertainties δ_j, that is $|\delta_j| \leq 1$. Following the results presented in [50], after some transformations, the system can be written in a new form Λ with extra inputs and outputs given in Eq. (8):

$$\begin{bmatrix} \dot{x} \\ \sigma \\ q_1 \\ \vdots \\ q_m \end{bmatrix} = \underbrace{\begin{bmatrix} A_0 & B_0 & E_1 & \ldots & E_m \\ C_0 & D_0 & F_1 & \ldots & F_m \\ G_1 & J_1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ G_m & J_m & 0 & \ldots & 0 \end{bmatrix}}_{\Lambda} \begin{bmatrix} x \\ r_1 \\ p_1 \\ \vdots \\ p_m \end{bmatrix}, \tag{8}$$

as exhibited in Fig. 2. The uncertain system is represented as a LFT around $\Lambda(s)$, namely $\sigma = F_L(\Lambda, \Delta)r_1$, where Δ maps $q \to p$ (see Fig. 3), and has the following block-diagonal structure:

$$\Delta = \{diag[\delta_1 I_{\rho_1}, \ldots, \delta_m I_{\rho_m}] : \delta_j \in \mathbb{R}\}. \tag{9}$$

The separation of the plant as in Fig. 3 is the first step in order to apply most of the robust control analysis and design techniques [20].

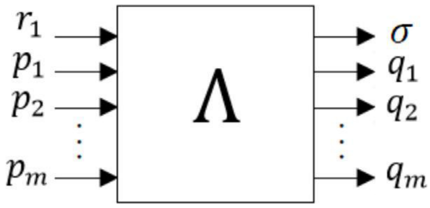

Fig. 2. Λ System.

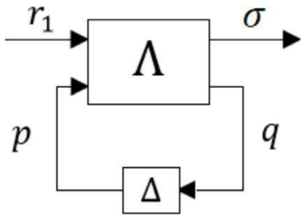

Fig. 3. Λ-Δ-Structure.

3.4. Definition and basic theorems for μ-analysis

Developed by Doyle [19], the structured singular value (SSV) is an efficient tool in robust control for checking necessary and sufficient conditions for robustness of stability and performance. Normally, the notation employed for SSV is μ or mu. On the other hand, this tool can only be used with the aid of computers, as it is necessary to perform numerical optimization routines [20]. Also, if one wants to apply commercial software implementations, like MATLAB®'s Robust Control Toolbox®, for SSV calculation, it is only possible for rational transfer functions matrices, which means that for ULTITDS, Padé or other approximations should be used in association with μ tools. The main results of the present paper, like Theorem 4, go in that direction, but taking a limit process.

Consider again the system in Fig. 3 and let us call the matrix Λ by M, which is the common name used in robust control literature. It is supposed that M is a matrix in which the controller was already included, that is, the upper LFT in Fig. 11 was performed [20]. This matrix can then be partitioned as:

$$M(s) = \begin{bmatrix} M_{11}(s) & M_{12}(s) \\ M_{21}(s) & M_{22}(s) \end{bmatrix} \tag{10}$$

and is called the closed loop nominal matrix, as it does not include the normalized uncertainties. In order to evaluate the robustness of stability, it is clear that only M_{22} is relevant, as it is the only part of M that connects to Δ.

Definition 2 (*[20,51]*)**.** Given a matrix M_{22} and a structured family of matrices Δ such that $\|\Delta\|_\infty \leq 1$, the structured singular value, also denoted by SSV or simply μ, is a real non-negative function defined by :

$$\mu_\Delta(M_{22}) \triangleq \frac{1}{min_\Delta\{\bar{\sigma}(\Delta)|\det(I - M_{22}\Delta) = 0\}}. \tag{11}$$

where M_{22} is a complex matrix, $\Delta = diag\{\Delta_i\}$ for $i \in \{1, \dots, s\}$ is a block diagonal matrix such that $\bar{\sigma}(\Delta_i) \leq 1$, where $\bar{\sigma}(A)$ denotes the maximum singular value of A. Some of the blocks of Δ may be repeated and some may be restricted to be real.

One can state the following theorem that provides a necessary and sufficient condition for robust stability using μ. This theorem was adapted from Skogestad and Postlethwaite [20].

Theorem 3. *Assume that the system $M_{22}(j\omega)$ is stable. Then the M_{22}-Δ-system in Fig. 4 is robustly stable for all allowed perturbations with $\bar{\sigma}(\Delta) \leq 1$, $\forall\omega$, if and only if:*

$$\mu_\Delta(M_{22}(j\omega)) < 1, \quad \forall\omega. \tag{12}$$

Proof. It can be found in Skogestad and Postlethwaite [20]. □

It can be shown that in order to evaluate the robustness of performance (that will be useful in Section 5) an additional uncertainty matrix as presented in Fig. 12 (that is, the matrix $\bar{\Delta}$) must be included [20] and in this case all the blocks of $M(s)$ play a role.

4. μ-Analysis for ULTITDS with Padé approximations: Main results

4.1. Necessary and sufficient condition for ULTITDS

In this section, it is presented the main results of the paper, that is Theorem 4 and the other theorems and lemmas used in its proof, which are contributions *per se*. An initial version for non-linear systems was published as a conjecture [26] (Conjecture

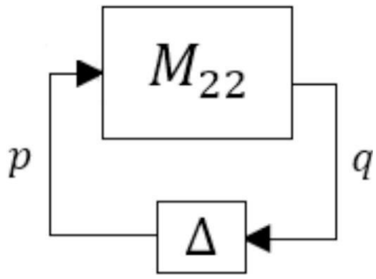

Fig. 4. M_{22}-Δ-Structure for robust stability analysis.

1), that is, without proof. In the present version, it is valid for ULTITDS and provides a sufficient condition for robust stability. In order to obtain the proof, μ-analysis, parametrized optimization theory and some theorems about the Padé approximation and its convergence are needed.

Theorem 4 (*Main Result*). *The system in Eq. (1) put in the form of Eq. (8) and Fig. 3, is robustly stable if, and only if:*

1. $\Lambda_{22,delay}(s)$ *is stable, and*
2. $\mu_\Delta(\Lambda_{22,k}(j\omega)) < 1, \quad \forall\omega, \forall k \in \mathbb{Z}^+$,

where:

- $\Lambda_{22,delay}(s) = G(sI - A_0 - F_\tau e^{-\tau s})^{-1}E$.
- $\mu_\Delta(\Lambda_{22,k}(j\omega)) = G(sI - A_0 - F_\tau[k,k])^{-1}E$.
- $[k,k]$ *is a Padé approximation for* $e^{-\tau s}$ *of kth order.*
- $G = [G_1, \dots, G_m]^T$, $E = [E_1, \dots, E_m]$ *are matrices from the decomposition* $A_j = E_j G_j$ *for* $j = 1, \dots, m$.

In order to prove this theorem, several auxiliary theorems and lemmas must be proved. Define the matrix function $\Phi(s) = sI - A_0 - F_\tau e^{-st}$. The elements of this matrix are *quasi-polynomials*, which is a subclass of *entire functions* [52].

Lemma 1. *Let be the sequence of maps* $\Phi_k(s) = sI - A_0 - F_\tau[k/k]$, *where* $k \in \mathbb{Z}_0^+$ *which also forms a sequence of rational matrices. This sequence of matrices is point-to-point and uniformly convergent to the matrix of holomorphic functions* $sI - A_0 - F_\tau e^{-\tau s}$.

Proof. Each entry of $F_\tau[k/k]$ is a rational function. By Theorem 2, this sequence is uniformly convergent to $F_\tau[k/k]e^{-\tau s}$, so the matrix sequence converges to $F_\tau e^{-st}$. Since $sI - A_0$ can be thought as a constant sequence (which always converges), and the sum of two convergent sequences of complex matrices always converges, then $\{\Phi_k(s)\}$ converges to $\Phi(s)$. The matrix sequence $\{F_\tau[k/k]_k\}$ is then formed by entries that are uniformly convergent. Using the ∞-norm, one has that:

$$\begin{aligned}&\|\Phi_k(s) - \Phi(s)\|_\infty = \|sI - A_0 - F_\tau[k/k] - sI + A_0 + F_\tau e^{-\tau s}\|_\infty = \\ &= \|F_\tau(e^{-\tau s} - [k/k])\|_\infty = \max_i(\sum_j |[F_\tau]_{ij}||[k/k] - e^{-\tau s}|) = \|F_\tau\|_\infty |e^{-\tau s} - [k/k]|.\end{aligned} \tag{13}$$

Since for every $\epsilon > 0$, there exists $K > 0$ such that for all $k > K$, $|[k/k] - e^{-\tau s}| < \epsilon$, one can always choose a $\epsilon_2 > 0$ such that there is a $\epsilon = \epsilon_2/\|F_\tau\|_\infty$ and a $K > 0$ such that for all $k > K$, one has:

$$\|\Phi_k(s) - \Phi(s)\|_\infty = \|F_\tau\|_\infty \epsilon < \epsilon_2,$$

then $\{\Phi_k(s)\}$ converges uniformly for the norm in question. Since the space of matrices is of finite dimension, then if a sequence converges in one norm, it converges in all other norms of the same space, which proves the lemma. □

Other theorems on convergence of Padé approximations, for example with different degrees of the polynomials $[k/m]$, could be used here without altering significantly the result. One needs now to define the sequence $\{\Phi_k^{-1}(s)\}$ and to know if it is uniformly convergent [21,46].

Lemma 2. *The sequence of rational matrices* $\{\Phi_k^{-1}(s)\}$ *is point-to-point convergent, except in a zero measured set of points.*

Proof. The sequence $\Phi_k(s)$ is convergent according to Lemma 1. Also, $f(A) = A^{-1}$ is continuous if and only if $A \in GL(\mathbb{C}, n)$, that is, it belongs to the set of non-singular matrices (remember that $GL(\mathbb{C}, n)$ is the group of all invertible $n \times n$ complex matrices [53]). Due to the fact that a sequence $\{f(x_k)\}$ is convergent if $\{x_n\}$ is convergent and if f is continuous [52,53], let us exclude the finite set of poles where $\Phi_k(s)$ is not defined. Due to the fact that the entries of the matrices $\Phi_k(s)$ are meromorphic, this set of points has also zero measure in $\mathbb{C}$, so every $\Phi_k(s)$ is defined almost everywhere in $\mathbb{C}$ (that is, in an open set). The function f obviously is defined in such open set and is continuous, then $\{\Phi_k^{-1}(s)\}$ converges in this set, and the result follows. □

Lemma 3. *The sequence of rational matrices $\{\Phi_k^{-1}(j\omega)\}$ is uniformly convergent almost everywhere.*

Proof. One must prove that for each $\epsilon_2 > 0$, there exists $K > 0$ such that for $k > K$, one has $\|\Phi_k^{-1}(s) - \Phi^{-1}(s)\|_\infty < \epsilon_2$. By using the identity $S^{-1} - T^{-1} = -S^{-1}(S-T)T^{-1}$, one has (in the open set described above, that is, almost everywhere):

$$\begin{aligned}
&\|\Phi_k^{-1}(s) - \Phi_\infty^{-1}(s)\|_\infty \leq \\
&\|(sI - A_0 - F_\tau[k/k])^{-1}\|_\infty \|F_\tau([k/k] - e^{-st})\|_\infty \|sI - A_0 - F_\tau e^{-st}\|_\infty = \\
&= |[k/k] - e^{-st}| \underbrace{\|(sI - A_0 - F_\tau[k/k])^{-1}\|_\infty \|F_\tau\|_\infty \|sI - A_0 - F_\tau e^{-st}\|_\infty}_{=\Gamma(s)}.
\end{aligned} \tag{14}$$

Let us assume $s = j\omega$, then there is a constant $\Lambda > 0$ such that $|\Gamma(j\omega)| < \Lambda$, which is a superior bound to the factors as indicated in Eq. (14). As $[k/k]$ converges uniformly to e^{-st}, then for each $\epsilon > 0$ there is a K such that for $k > K$, one has $|[k/k] - e^{-st}| < \epsilon$, then, for each $\epsilon_2 > 0$ there is K such that for $k > K$, one has:

$$\|\Phi_k^{-1}(s) - \Phi^{-1}(s)\|_\infty \leq |[k/k] - e^{-st}|\Lambda < \epsilon\Lambda = \epsilon_2, \tag{15}$$

which concludes the proof. □

Lemma 4. *The sequence $\{\Lambda_{22,k}(s)\} = G\{\Phi_k^{-1}(s)\}E$ converges uniformly almost everywhere to $\Lambda_{22,delay} = G(sI - A_0 - Fe^{-\tau s})^{-1}E$.*

Proof. Since $\Phi_k^{-1}(s)$ is uniformly convergent almost everywhere in $\mathbb{C}$ and if it is just multiplied by constants on both sides, then the sequence continues to uniformly converge, and the lemma is proved. □

As can be seen in Theorem 4, one has to define a sequence of functions given by:

$$\mu_k(j\omega) = \{\mu_\Delta(\Lambda_{22,k}(j\omega))\}, \tag{16}$$

that is, one has to deal, for $k \in \mathbb{Z}_0^+$, with an optimization problem with restriction for some values of $\omega \in R^+$, that is, one is searching for Δ^*, corresponding to the minimum of $\bar{\sigma}(\Delta^*)$ with the restriction $\det(I - \Lambda_{22,k}\Delta(j\omega)) = 0$. It is therefore an optimization problem with restriction (see Definition 2). According to Doyle [19], those functions are continuous. Unfortunately, this type of optimization is complex to solve. From Doyle [19], the optimization problem:

$$min_{D\in D_\Delta}\bar{\sigma}(D\Lambda_{22,k}(j\omega)D^{-1}),$$

where D_Δ is the set of matrices that commute with Δ, is convex, which means that there is a single global minimum D^* and efficient algorithms to solve numerically this problem. Then, let us define the following sequence:

Definition 3 (*Upper Bound Sequence*)**.** The sequence of functions given by:

$$m_k(j\omega) = min_{D(\omega)\in D_\Delta}\bar{\sigma}(D(\omega)\Lambda_{22,k}(j\omega)D(\omega)^{-1}), \tag{17}$$

where D_Δ is the set of matrices that commutes with Δ, is an upper bound sequence of the sequence in Eq. (16).

Then $m_k(j\omega) \geq \mu_k(j\omega)$ for $k \in \mathbb{Z}_0^+$. To get conclusions about this sequence of functions, the following theorem is important:

Theorem 5. *If $H(j\omega)$ is an arbitrary transfer function which is continuous in $s = j\omega$, the function $m(j\omega)$ given by:*

$$m(j\omega) = min_{D\in D}\bar{\sigma}(DH(j\omega)D^{-1}) = min_{D\in D}\sqrt{\lambda_{max}(DH(j\omega)(D^\dagger D)^{-1}H^\dagger(j\omega)D^\dagger)}, \tag{18}$$

that is, a function given by the optimization on the parameter D, is continuous in $[0, \infty)$

Proof. Let us define the function $F(D, j\omega) = DH(j\omega)(D^\dagger D)^{-1}H^\dagger(j\omega)D^\dagger$ that is obviously continuous in ω. As the eigenvalues are continuously dependent on the matrix entries, the function $\lambda_{\max}(F(D, j\omega))$ is also continuous in ω (see [54] p. 497). Finally, as the square root is continuous for non-negative values, $\bar{\sigma}(H(j\omega)) = \sqrt{\lambda_{\max}(F(D, j\omega))}$ is continuous in $[0, \infty)$. Supposing that the optimization problem in Eq. (18) is such that the set of decision values $G(\omega) = D_\Delta$ is a compact subset of D_Δ (which is reasonable to assume, as computers cannot store infinities) and the set valued function $G(\omega)$ is also continuous (as it is constant), then applying Theorem 1 (Berge's Continuous Maximum Theorem), the function $m(j\omega)$ must be continuous. □

Corollary 1. *The sequence of functions in Eq. (17) has only continuous functions in $[0, \infty)$.*

Then, each function in the sequence presented in Eq. (17) is continuous $(0, \infty)$. Now remains the task to prove that the sequence in Eq. (17) converges. The following theorem is then important.

Theorem 6. *The function of matrices:*

$$u(X) = min_{D\in D}\bar{\sigma}(DXD^{-1}),$$

is continuous in the parameters X.

Proof. It is well known that $\bar{\sigma}(DXD^{-1}) = \sqrt{\lambda_{max}(DX(D^{\dagger}D)^{-1}X^{\dagger}D^{\dagger})}$ is a continuous function on D and X, that is, the eigenvalues depend continuously on the matrix entries (see [54] p. 497). Let us suppose that the values of D used in the optimization belong to a compact set for each possible X, which is reasonable, as the possible matrices D represented in a computer should have limited values in their entries. Let us call this compact set $KD_{\Delta} \subset D_{\Delta}$. The set-valued function $G(X) = KD_{\Delta}$ is continuous, as it is constant. Then the set-valued function $G(X)$ is continuous and compact-valued. Then, by applying again the Berge's continuous maximum value theorem, the result follows. □

Finally, one can prove that the sequence in Eq. (17) is convergent:

Theorem 7. *The sequence (17) converges.*

Proof. By Lemma 4, the sequence $\{\Lambda_{22,k}(s)\}$ converges almost everywhere. As the function $u(X)$ is continuous in X by Theorem 6 and due to the fact that a sequence $\{f(x_k)\}$ is convergent if $\{x_n\}$ is convergent and if f is continuous, then there is an open set in which f is continuous, then the result follows. □

Theorem 4 can then be proved:

Proof of Theorem 4. The system in Eq. (1) is in the form presented in Eq. (7) of Section 3.3. The delay term can be inserted later. Then, it can be written as in Eq. (8):

$$\begin{bmatrix} \dot{x} \\ \sigma \\ q_1 \\ \vdots \\ q_m \end{bmatrix} = \underbrace{\begin{bmatrix} A_0 & B & E_1 & \dots & E_m \\ C & 0 & 0 & \dots & 0 \\ G_1 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ G_m & 0 & 0 & \dots & 0 \end{bmatrix}}_{\Lambda} \begin{bmatrix} x \\ r_1 \\ p_1 \\ \vdots \\ p_m \end{bmatrix}, \tag{19}$$

with $F_j = J_j = 0$ and $A_j = E_j G_j$, where $A_j \in \mathbb{R}^{n \times n}$ (of $\Delta A = \sum_{j=1}^{m} A_j \delta_j$), $E_j \in \mathbb{R}^{n \times \rho_j}$, $G_j \in \mathbb{R}^{\rho_j \times n}$, for $j = 1, 2, \dots, m$, and $\rho_j = rank[A_j]$, $B_0 = B$, $C_0 = C$ and $D_0 = 0$. This can be done as the system has all parameters real. Developing the Eq. (19) and inserting the delay, one has:

$$\begin{cases} \dot{x} = & A_0 x + Br_1 + Ep + F_{\tau} x(t-\tau), \\ \sigma = & Cx, \\ q = & Gx. \end{cases} \tag{20}$$

Applying the Laplace transform, after some algebraic manipulation, one gets:

$$X(s) = (sI - A_0 - F_{\tau}e^{-\tau s})^{-1} Br_1(s) + (sI - A_0 - F_{\tau}e^{-\tau s})^{-1} Ep(s).$$

and substituting $X(s)$ into the second and third equations in Eq. (20), one has:

$$\sigma(s) = C(sI - A_0 - F_{\tau}e^{-\tau s})^{-1} Br_1(s) + C(sI - A_0 - F_{\tau}e^{-\tau s})^{-1} Ep(s)$$

and

$$q(s) = G(sI - A_0 - F_{\tau}e^{-\tau s})^{-1} Br_1(s) + G(sI - A_0 - F_{\tau}e^{-\tau s})^{-1} Ep(s).$$

After some manipulation, one has:

$$\Lambda_{delay} = \begin{bmatrix} \Lambda_{11,delay}(s) & \Lambda_{12,delay}(s) \\ \Lambda_{21,delay}(s) & \Lambda_{22,delay}(s) \end{bmatrix} = \begin{bmatrix} C(sI - A_0 - F_{\tau}e^{-\tau s})^{-1} B & C(sI - A_0 - F_{\tau}e^{-\tau s})^{-1} E \\ G(sI - A_0 - F_{\tau}e^{-\tau s})^{-1} B & G(sI - A_0 - F_{\tau}e^{-\tau s})^{-1} E \end{bmatrix}, \tag{21}$$

where the ULTITDS relating r_1 to σ is given by the lower LFT $\sigma = F_L(\Lambda_{delay}, \Delta) r_1$ - see Eq. (5) - where $F_L(\Lambda_{delay}, \Delta) = \Lambda_{11,delay} + \Lambda_{12,delay}\Delta\Lambda_{21,delay}(I - \Lambda_{22,delay}\Delta)^{-1}$ (see Fig. 3).

Remark 1. Note that the only term required for the stability analysis is $\Lambda_{22,delay}$.

From now on, it will be worked with Padé approximations where $M = L$, i.e. the degree of the numerator is equal of the denominator in Eq. (4). When the exponential is replaced by a kth order Padé approximation, a sequence of rational matrices is obtained, which is:

$$\Lambda_{22,k}(j\omega) = G(j\omega I - A_0 - F_{\tau}[k/k])^{-1} E. \tag{22}$$

One has already proved, in Lemma 4, that this sequence converges uniformly. By Theorem 3, the system $\sigma = F_L(\Lambda_k(s), \Delta) r_1$ is robustly stable if and only if:

$$\mu_k(j\omega) = \mu_{\Delta}(\Lambda_{22,k}(j\omega)) < 1, \tag{23}$$

where Δ is a block diagonal matrix of uncertainty, and if $1 > m_k(j\omega) > \mu_k(j\omega)$, the condition is also satisfied. One also already proved, in Theorem 7, that the sequence $m_k(j\omega)$ converges to a function that one could call $m_{delay}(j\omega)$, and due to the order limit theorem, as $m_k(j\omega) > \mu_k(j\omega)$, then $m_{delay}(j\omega) > \mu_{delay}(j\omega) = \mu_{\Delta}(\Lambda_{22,delay}(j\omega))$, which concludes the proof. □

Table 2
Convergence of $\|m_k\|_\infty$ for $\tau = 1.5$.

k	1	2	3	4	5	6	7
$\| m_k\|_\infty$	0.8813	0.5045	0.8234	0.8240	0.8240	0.8240	0.8240

Remark 2. The reader may ask whether the simple fact that $1 > m_k(j\omega) > \mu_k(j\omega)$ is not sufficient to guarantee that both sequences converge. In fact, it is, but we cannot assume that 1 is a superior bound *a priori*. On the contrary, if one is using an algorithm like the DK interaction for control design, in which calculations are done mainly for $m(j\omega) > 1$, such restriction is not acceptable, as there would be no guarantee of the existence of $m_{delay}(j\omega)$ during the calculations. In this paper, it was proved that $m_{delay}(j\omega)$ and $\mu_{delay}(j\omega)$ are well defined even if no bound in $m_k(j\omega)$ is assumed and the space of superior bounds to μ of ULTITDS system (and its Padé approximations) forms a Banach space.

Remark 3. In practical terms, one only checks the condition $\mu(\Lambda_{22,k}(j\omega)) < 1$ for a finite subset of indexes $k_1 < \cdots < k_p$. Due to the fact that $m_{delay}(j\omega) > \mu_{delay}(j\omega)$, if it is clear that in the calculations $m_k(j\omega)$ will not cross 1, it is almost certain that the ULTITDS is robustly stable. In a future paper, the bounds on the error approximation for e^{-Ts} presented in [46] will be used to estimate the velocity of convergence of sequence $m_k(j\omega)$.

Remark 4. In order to verify that $\Lambda_{22,delay}(s)$ is stable (condition 1 of Theorem 4), roots of quasi-polynomials must be found, which is a well known technique.

4.2. Examples of application and comparison with other methods

In the following, examples are given in which the results presented in Section 4.1 are applied for robustness analysis. First of all, the system originally in the form of Eq. (1) must be put in the form of Eq. (20). The term $\Lambda_{22,delay}(s)$ can then be calculated (that is the nominal term for $\delta_i = 0$) and its stability is analysed. In order to guarantee that the stability persists in the presence of the uncertainties $|\delta_i| \le 1$, the analysis of the structured singular value sequence $\mu_k(j\omega) = \mu_\Delta(\Lambda_{22,k}(j\omega))$ of $\Lambda_{22,k}(j\omega)$ as given in Eq. (22) is performed. The uncertainty matrix is given by $\Delta = \mathrm{diag}(\delta_1, \dots, \delta_p)$.

This sequence of functions can only be determined numerically using the MATLAB® function **mussv** (or a similar one) which only gives the superior bound sequence $m_k(j\omega)$ in graphic form. If $m_k(j\omega) < 1$ for a reasonable number of terms in the series and a convergence pattern is observed, the robustness of stability, that is, in the presence of uncertainties, is guaranteed. In practical situations, the nominal system, that is, without uncertainties, will be a closed-loop system in which the controller was designed such that the nominal closed-loop system is stable. The method, in this case, would be applied to check robustness of stability.

Example 1. Consider the system with two parametric uncertainties such that $|\delta_j| \le 1$, for $j = 1, 2$, null inputs, and one constant delay τ:

$$\begin{cases} \dot{x}_1 = -1.4x_1 + 1.2\delta_1 x_2 - 2x_1(t-\tau) + r_1, \\ \dot{x}_2 = -1.1x_2 + \delta_2 x_1 + 0.1x_2(t-\tau) + r_1. \end{cases}$$

In this example, one has:

$$A_0 = \begin{pmatrix} -1.4 & 0 \\ 0 & -1.1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad F_\tau = \begin{pmatrix} -2 & 0 \\ 0 & 0.1 \end{pmatrix},$$

$$\Delta A = \begin{pmatrix} 0 & 1.2\delta_1 \\ \delta_2 & 0 \end{pmatrix} = \sum_{j=1}^{2} A_j \delta_j, \text{ where } A_1 = \begin{pmatrix} 0 & 1.2 \\ 0 & 0 \end{pmatrix} \text{ and } A_2 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}.$$

As $A_1 = E_1 G_1$ with $\rho_1 = 1$, $E_1 = \begin{pmatrix} 1.2 \\ 0 \end{pmatrix}$ and $G_1 = \begin{pmatrix} 0 & 1 \end{pmatrix}$; and $A_2 = E_2 G_2$, $\rho_2 = 1$ with $E_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ and $G_2 = \begin{pmatrix} 1 & 0 \end{pmatrix}$, it is possible to calculate the matrices $\Lambda_{22,k}$ and its superior bounds to the SSV and apply Theorem 4. Assume that $\tau = 1.5$. In Fig. 5 it is shown graphs of $m_k(j\omega)$ for different values of k, which were obtained by the **mussv** function in MATLAB®'s Robust Control System Toolbox. The uncertainty matrix, which is required by the function, is $\Delta = \mathrm{diag}(\delta_1, \delta_2)$. One can see the convergence of the sequence of functions $m_k(j\omega)$, which is the upper bound for $\mu_k(j\omega)$. It is clear that the sequence of functions converges for values below 1.

The maximum value for $m_k(j\omega)$, which is the norm $\|m_k\|_\infty$ is presented in Table 2. It is clear the convergence of the sequence, which assures robustness of stability. This method can also be used to search for the maximum values of τ and/or the uncertainties δ_j such that robustness of stability is achieved, which is done in the next example.

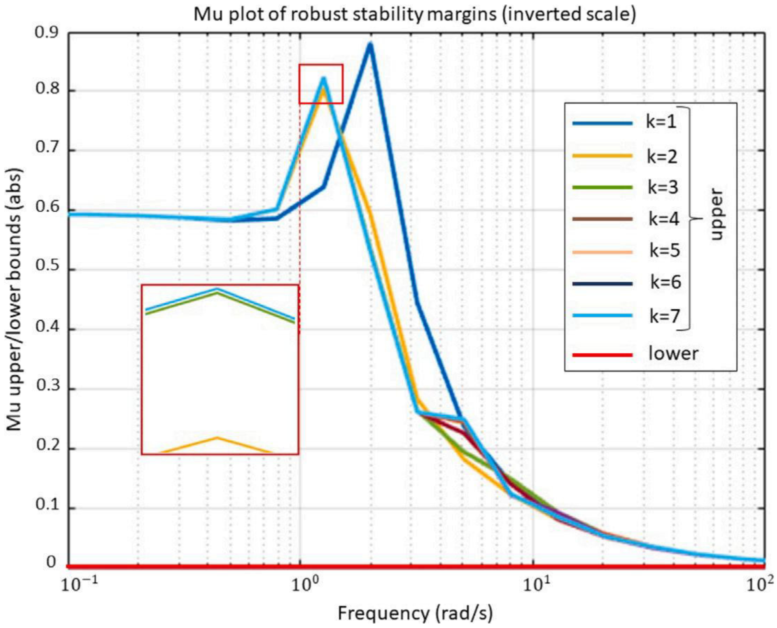

Fig. 5. μ-Analysis for increasing values of k.

Table 3
τ_{max} and Padé approximation order.

Order (k)	1	2	3	4	...	10	...	20
τ_{max}	1.979	1.698	1.644	1.642	...	1.642...	...	1.642...
$\lVert m_k \rVert_\infty$	0.999	0.999	0.9986	0.9987	...	0.987	...	0.9987

Example 2. Let us find, for the same system in Example 1, the maximum value τ_{max} of the time-delay in order to have robustness of stability. One knows that the maximum value of τ is achieved if $\lVert m_k \rVert_\infty$ asymptotically approaches 1 from below (or, on the other hand, no robustness of stability exists). For each k, that is, order of the Padé approximation, the maximum value τ_{max} was found and put in Table 3. Another sequence with index k appears that converges to the maximum allowed delay. Note that for first and second order Padé approximations, the maximum delay value such that one has robustness of stability is considerably higher than the limit value. One can also argue whether the sequence $\tau_{\max}(k)$ is always convergent (which apparently is), but it will be subject to future research.

In order to validate the predictions, a simulation is performed in Simulink for validating the method. Fig. 6 shows the time domain simulation where some values of δ in the interval $[-1, 1]$ are varied and $\tau = 1.642$. On the other hand, for values of τ greater than 1.642, shown in Fig. 7, it is observed that the system is unstable.

Example 3 (*Comparison*)**.** This example presents a comparison of the method of Theorem 4 developed in this paper with the LMI methods of Fridman and Shaked [55,56]. Consider the system:

$$\begin{cases} \dot{x}_1 = -0.12x_2 + 0.42\delta x_2 - 0.1x_1(t-\tau) - 0.35x_2(t-\tau), \\ \dot{x}_2 = x_1 - 0.465x_2 - 0.035\delta x_2 + 0.3x_2(t-\tau). \end{cases}$$

In this example, one has:

$$A_0 = \begin{pmatrix} 0 & -0.12 \\ 1 & -0.465 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad F_\tau = \begin{pmatrix} -0.1 & -0.35 \\ 0 & 0.3 \end{pmatrix},$$

$$\Delta A = A_1 = \begin{pmatrix} 0 & 0.42\delta \\ 0 & -0.035\delta \end{pmatrix} = E_1 G_1,$$

where $E_1 = \begin{pmatrix} 0.42 \\ -0.035 \end{pmatrix}$ and $G_1 = \begin{pmatrix} 0 & 1 \end{pmatrix}$. Once the matrices are obtained A_0, F_τ, E and G, it is possible to apply Theorem 4.

For this example, in [55] (Theorem 2) a $\tau_{max} = 0.782$ was found; and in [56] (Theorem 1) was obtained a $\tau_{max} = 0.863$. Now, applying Theorem 4 of the present paper, one gets $\tau_{max} = 0.898$. This shows that for this example the method is less conservative than others in the literature. Furthermore, according to the Simulink's simulations, Figs. 8 and 9, it is even possible to say that the method is practically non-conservative: Fig. 8 shows instability when $\tau_{max} = 0.899$ is used; on the other hand, Fig. 9 shows the simulation with $\tau_{max} = 0.898$ and the system is stable.

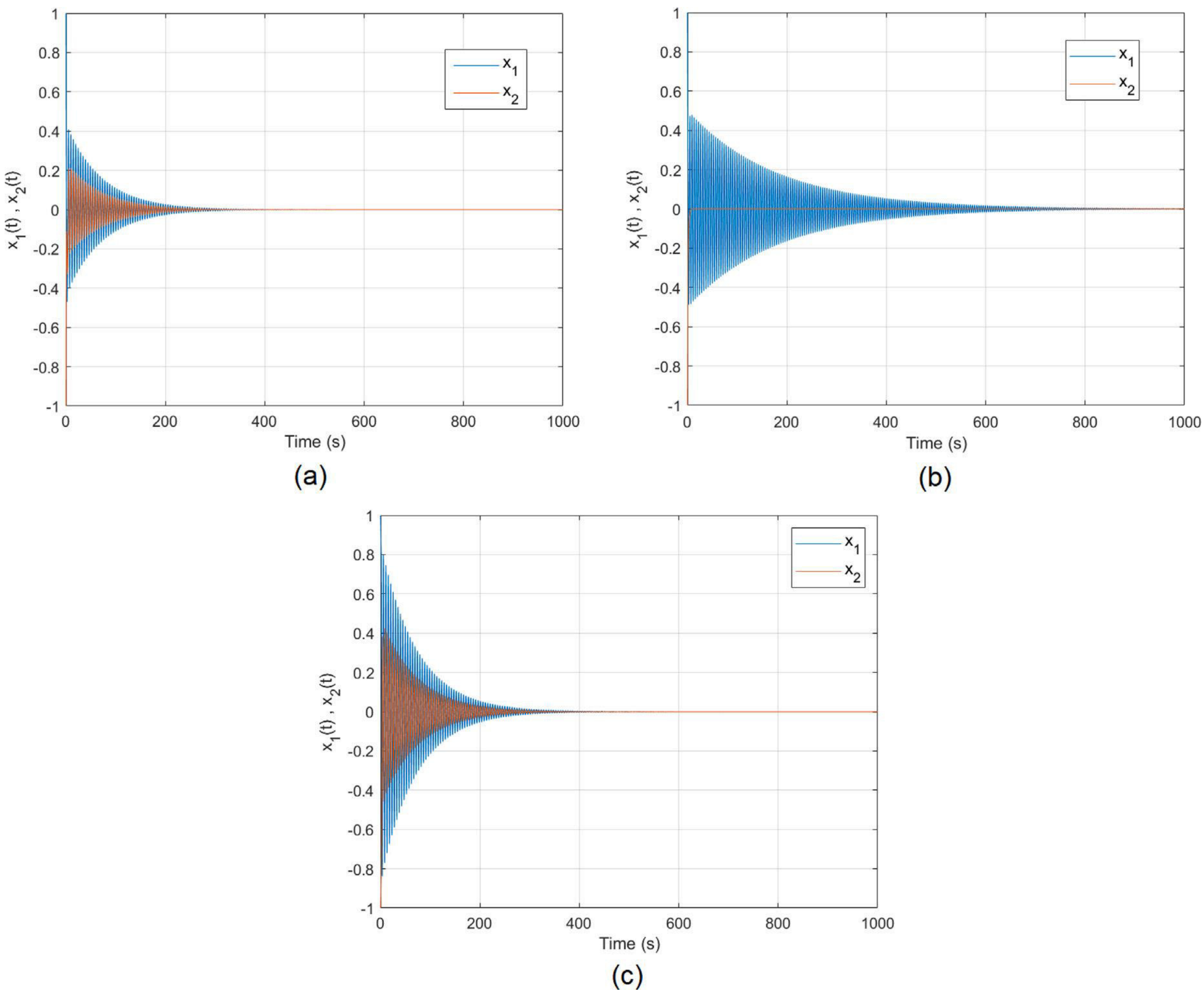

Fig. 6. Robust stability for different values of δ: (a) $\delta_1 = \delta_2 = 1$; (b) $\delta_1 = \delta_2 = 0$; (c) $\delta_1 = \delta_2 = -1$.

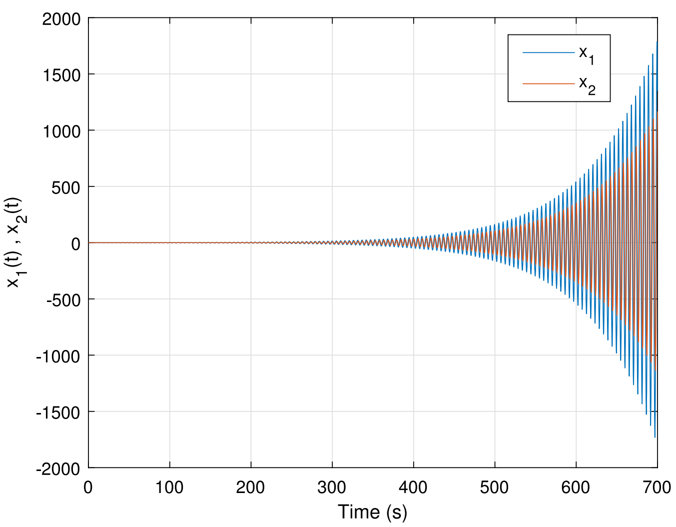

Fig. 7. Unstable system with $\tau = 1.7$.

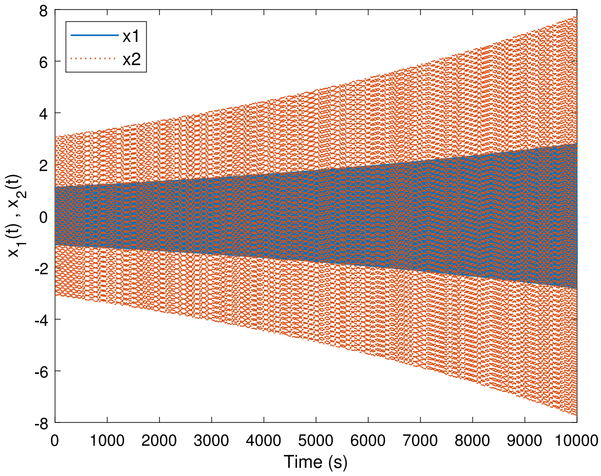

Fig. 8. Time response for $\tau_{max} = 0.899$: unstable system.

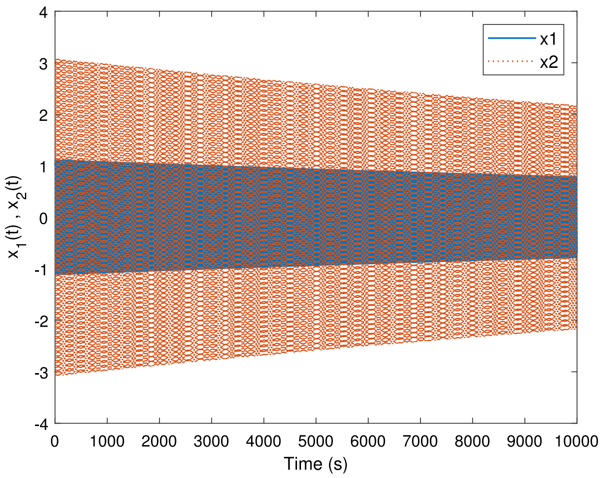

Fig. 9. Time response for $\tau_{max} = 0.898$: stable system.

Another comparative analysis is necessary now in which the difficulty of application of the method of this paper (that is, based in Theorem 4) is compared to others in the literature. First of all, there are few methods in the literature that deal with the same problem, as far as the authors know. The method in paper [56], which was used in the preceding example, despite the fact of being more conservative, is based in Linear Matrix Inequalities (LMI). There are efficient algorithms for solving LMI, but tools like SSV, developed in the eighties, are better known, mainly for the practitioners. The same can be said about other methods which are LMI-based, like [57]. In the particular case of this last paper, uncertainties are not considered. The paper [58] uses SSV, but, despite of the claim that no conservatism is introduced, Padé approximations of low order are introduced, which can be conservative in general. All the simulations of the present paper's method took around three minutes to complete in a Laptop with Intel® i7 processor with 32 GB of RAM.

5. μ-Synthesis for uncertain linear time invariant TDS systems

It is also possible to use the results of this paper in the design of feedback controllers for ULTITDS via DK-iteration, that is, μ-synthesis [20]. In such case, one is interested in designing the controller's matrix of transfer functions $K(s)$ in Fig. 10, which is a closed-loop system in which the system in Fig. 1 is the plant. The DK-iteration is a method to design robust controllers in which while robustness is not achieved, two steps are performed: **step K** and **step D**. In **step K**, a preliminar controller is designed by H_∞ minimization, and in **step D** another minimization is done in which an upper bound to $\mu(M)$ is obtained, where M is like in Eq. (10) and Fig. 12 [20]. The robustness pursued here is then the robustness of performance. One knows that robustness was not achieved while (the superior limit to SSV) $m(j\omega) > 1$ for some ω. Algorithm 1 presented below realizes this process [20]. The **step K** involves a H_∞-synthesis step, that is, a controller with the classical separation structure of an observer and a state feedback is produced.

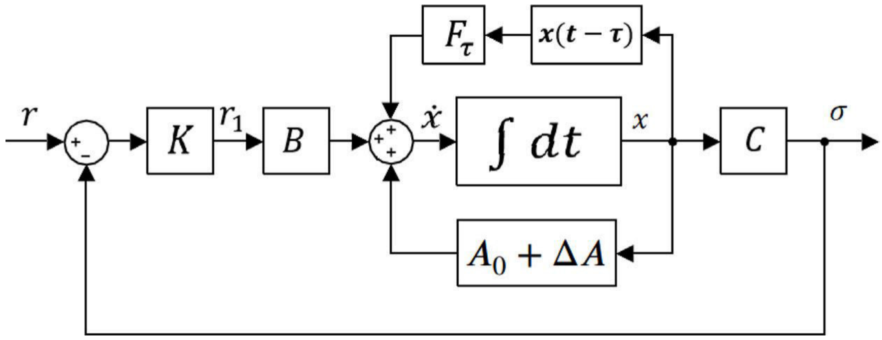

Fig. 10. Closed-Loop ULTITDS.

5.1. Main results for the design algorithm

Examining the Algorithm 1, the closed-loop nominal system $M(s)$ is then necessary. One must then guarantee that the closed-loop system in Fig. 10, that is the plant + controller, could be put in the form of Eq. (1). Lemma 5 is then important in this regard.

Lemma 5. *The closed-loop system in Fig. 10 where the controller $K(s)$ is obtained by DK-iteration has a state-space representation of the form:*

$$\dot{\bar{x}}(t) = (\bar{A}_0 + \Delta\bar{A})\bar{x}(t) + \bar{B}r(t) + \bar{F}_\tau \bar{x}(t-\tau), \tag{24}$$

where $\bar{x}(t) = [x(t)\ \hat{x}(t)]^T$.

Proof. As shown in Algorithm 1, in each iteration a H_∞-synthesis step is performed, which produces for the controller $K(s)$ a well known separation structure of a state feedback $u = F_\infty \hat{x}(t)$ and observer:

$$\dot{\hat{x}}(t) = A_\infty \hat{x}(t) + Bu(t) + Z_\infty L_\infty \sigma(t) \tag{25}$$

where A_∞ is stable and $\sigma = Cx$ is the plant output [20]. As the ULTITDS plant is of the form:

$$\dot{x}(t) = (A_0 + \Delta A)x(t) + Bu(t) + F_\tau x(t-\tau), \tag{26}$$

one can write:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{\hat{x}}(t) \end{bmatrix} = \begin{bmatrix} A_0 + \Delta A & BF_\infty \\ BF_\infty + Z_\infty L_\infty C & A_\infty \end{bmatrix} \begin{bmatrix} x(t) \\ \hat{x}(t) \end{bmatrix} + \begin{bmatrix} F_\tau & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x(t-\tau) \\ \hat{x}(t-\tau) \end{bmatrix} + \begin{bmatrix} I \\ 0 \end{bmatrix} r \tag{27}$$

which concludes the proof. □

DK-iteration must guarantee robustness of performance, and not only robustness of stability, for the closed-loop system in Fig. 10 [20]. The structured singular value, as defined in Definition 2 can be used in the following way [20]:

Theorem 8. *Consider the system in Fig. 12, and assume that the nominal system M is (internally) stable. Then the system has robustness of performance if, and only if:*

$$\mu_{\tilde{\Delta}}(M(j\omega)) < 1, \quad \forall\omega, \tag{28}$$

where μ is computed with respect to the matrix:

$$\tilde{\Delta} = \begin{bmatrix} \Delta & 0 \\ 0 & \bar{\Delta} \end{bmatrix}, \tag{29}$$

and $\bar{\Delta}$ with the same dimensions as $F_L(M,\Delta)^T$.

Proof. It can be found in Skogestad and Postlethwaite [20]. □

Now, the complete matrix $M(s)$ must be used, and not only $M_{22}(s)$. The uncertainty matrix $\bar{\Delta}$ is fully complex (that is, all the entries are complex functions) and $\|\bar{\Delta}\|_\infty \leq 1$. If one refers to Fig. 11, one can give a simple interpretation: the vector of external signals $w(s)$, in the robustness of performance analysis, is given by $\bar{\Delta}(s)z(s)$, which is the second feedback loop in Fig. 12. It represents the worst disturbances/reference that can occur, which should use information of the performance signal $z(t)$ in order to make it goes to infinity, which is the worse performance and is equivalent to make the system in Fig. 13 unstable. The controller's designer goal would be to avoid this instability by designing an appropriate controller $K(s)$, which in principle is possible as she knows the model of w (that is the family of matrices $\bar{\Delta}(s)$) [20].

As the plant is of the form in Fig. 1, it can be put in the form of Fig. 3. The closed-loop system is then of the form in Fig. 13. In order to be useful for the Algorithm 1, this closed-loop system must be put in the form of Fig. 12. Theorem 9 is then necessary:

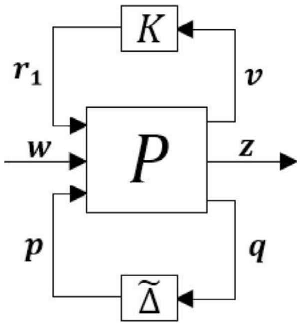

Fig. 11. General control system.

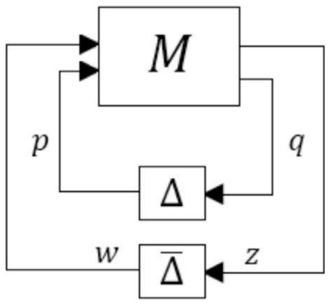

Fig. 12. M-Δ-Structure for robust performance analysis.

Theorem 9. *The closed-loop system in Fig. 13 can be put in the form of Fig. 11 where the augmented plant $P(s)$ is given by:*

$$P(s)=\left[\begin{array}{c|c|c} \Lambda_{11,delay} & -I & -\Lambda_{12,delay} \\ \hline \Lambda_{11,delay} & I & \Lambda_{12,delay} \\ \hline \Lambda_{21,delay} & 0 & \Lambda_{22,delay} \end{array}\right], \tag{30}$$

and in the form of Fig. 12, where $M(s)$ is given by:

$$M(s)=\left[\begin{array}{c|c} I-\Lambda_{11,delay}K(I-\Lambda_{11,delay}K)^{-1} & \Lambda_{12,delay}-\Lambda_{11,delay}K(I-\Lambda_{11,delay}K)^{-1}\Lambda_{12,delay} \\ \hline -\Lambda_{21,delay}K(I-\Lambda_{11,delay}K)^{-1} & \Lambda_{12,delay}-\Lambda_{21,delay}K(I-\Lambda_{11,delay}K)^{-1}\Lambda_{12,delay} \end{array}\right]. \tag{31}$$

Proof. If one considers the augmented plant of Fig. 13 as the one having as inputs the signals w, r_1, p and as outputs the signals z, q and v, it can be represented by:

$$\begin{bmatrix} v \\ z \\ q \end{bmatrix}=\begin{bmatrix} \Lambda_{11,delay} & -I & -\Lambda_{12,delay} \\ \Lambda_{11,delay} & I & \Lambda_{12,delay} \\ \Lambda_{21,delay} & 0 & \Lambda_{22,delay} \end{bmatrix}\begin{bmatrix} r_1 \\ w \\ p \end{bmatrix}+\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}r, \tag{32}$$

which is in the form of Fig. 11 with K and $\tilde{\Delta}$ removed. Consider now that the controller $K(s)$ is inserted and $r_1 = K(s)v$. After substituting in the first line in Eq. (32), one has:

$$v=-(I-\Lambda_{11,d}K)^{-1}w-(I-\Lambda_{11,d}K)^{-1}\Lambda_{12,d}Kp+(I-\Lambda_{11,d}K)^{-1}r,$$

and as $r_1 = Kv$, after substituting in the second and third lines in Eq. (32), one has that the $M(s)$ system is given by:

$$\begin{bmatrix} z \\ q \end{bmatrix}=\left[\begin{array}{c|c} I-\Lambda_{11,delay}K(I-\Lambda_{11,delay}K)^{-1} & \Lambda_{12,delay}-\Lambda_{11,delay}K(I-\Lambda_{11,delay}K)^{-1}\Lambda_{12,delay} \\ \hline -\Lambda_{21,delay}K(I-\Lambda_{11,delay}K)^{-1} & \Lambda_{12,delay}-\Lambda_{21,delay}K(I-\Lambda_{11,delay}K)^{-1}\Lambda_{12,delay} \end{array}\right]\begin{bmatrix} w \\ p \end{bmatrix}+$$
$$+\left[\begin{array}{c} \Lambda_{11,delay}K(I-\Lambda_{11,delay}K)^{-1} \\ \hline \Lambda_{11,delay}K(I-\Lambda_{11,delay}K)^{-1} \end{array}\right]r, \tag{33}$$

which concludes the proof. □

5.2. Examples of application and discussion

The Algorithm 1 together with Theorems 4 and 9 establish a practical methodology for controller design via DK-iteration for ULTITDS. After putting the plant + controller in the form of Fig. 12 (application of Theorem 9), for each Padé approximation of e^{-Ts} (in general only seven approximations are necessary) the Algorithm 1 is applied. Padé approximations are necessary as

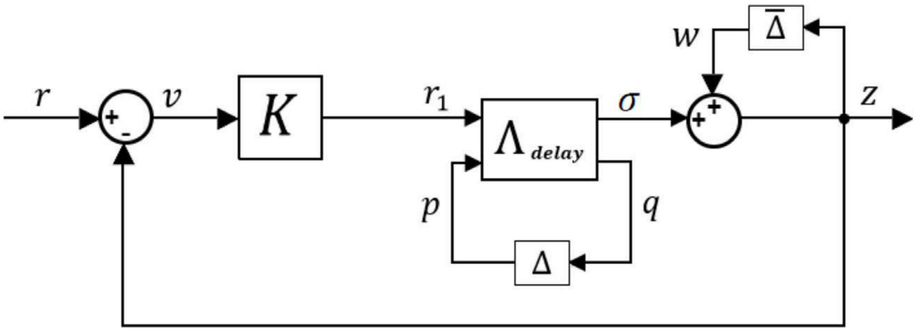

Fig. 13. Λ-Δ-$\bar{\Delta}$-Structure in closed-loop with controller.

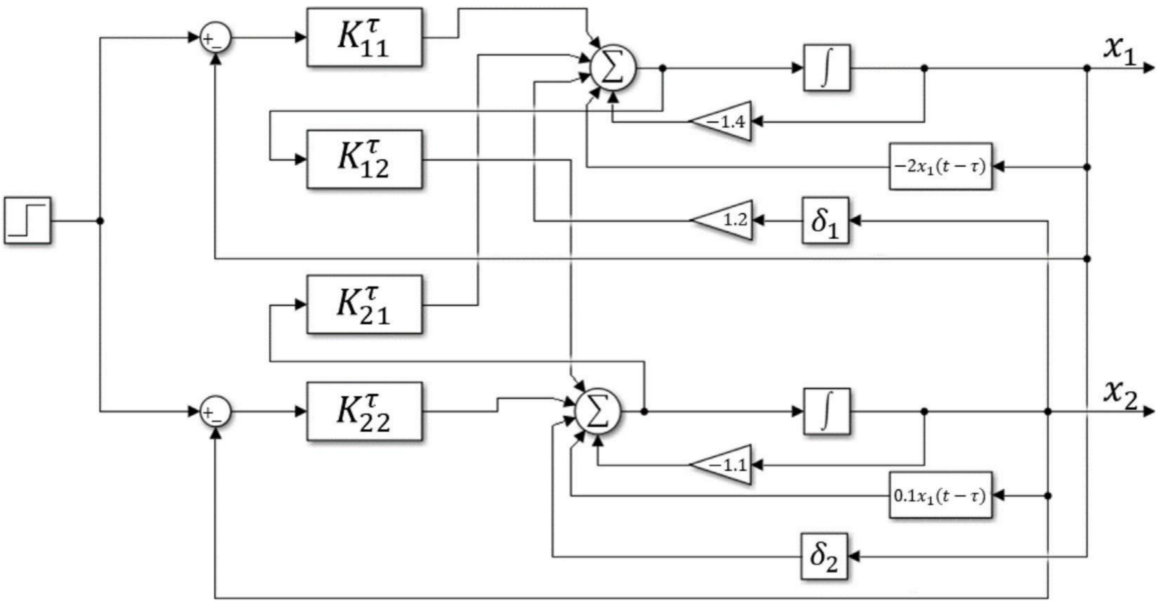

Fig. 14. ULTITDS in closed-loop.

the MATLAB® functions only works with rational matrices of transfer functions. For each Padé approximation, a controller $K_k(s)$ will be obtained for $\|m_k\|_\infty$ very close to one. A truncated sequence of controllers that guarantees robustness of performance (and consequently robustness of stability) $K_k(s)$ is obtained.

Algorithm 1 DK-iteration.

Result: A stabilizing controller $K(s)$ that guarantees robustness of performance.
Find the nominal augmented plant $P(s)$ as in Fig. 11 and the structure of matrix $\tilde{\Delta}$.
Start with an initial guess for D, usually set $D = I$.
WHILE ($m(j\omega) > 1$) or a pre-specified maximum iteration number is not reached
Step K: Synthesize a $\mathcal{H}_\infty$-controller $K(s)$, that is, solve $min_K ||DF_U(P,K)D^{-1}||_\infty$ with fixed $D(s)$. Use Matlab® function $[K, \gamma, \text{Info}] = \text{hinfsyn}(DPD^{-1})$
Calculate $M(s)$ with the upper LFT of Fig. 11
Step D: Find $D(j\omega)$ that minimizes $\bar{\sigma}(DMD^{-1}(j\omega))$ in each frequency, with fixed M. Use Matlab® function mussvunwrap(Info)
Adjust the magnitude of each element of $D(j\omega)$ to a stable and minimum phase transfer function $D(s)$. Use Matlab® function fitfrd
Calculate $m(j\omega)$, where $m(j\omega)$ is equal to upper bound of μ. Use Matlab® function mussv(frdModel,BlockStructure)

Example 4. Consider again the plant in Example 1. One wants to design a closed-loop MIMO controller $K^\tau(s)$ for this plant, so that the closed-loop system is the one in Fig. 14. One wants a steady-state error less than 10^{-5} and a settling-time less than 10^{-3} for unit step response. Those specifications must be translated in weight matrices that ultimately will be included in the augmented plant $P(s)$. Algorithm 1 were executed for Padé approximation till third order, in which convergence was secured.

Simulink is used to simulate the system of Fig. 14 with unit step input and with variations of δ_1 and δ_2. In Figs. 15–17 it is presented the step response in open-loop for different δ. The closed-loop step response is presented in Fig. 18 for the same δ's (i.e, $\delta_1 = \delta_2 = -1$, $\delta_1 = \delta_2 = 0$ and $\delta_1 = \delta_2 = 1$). One can see that the performance specifications are satisfied for all values of δ,

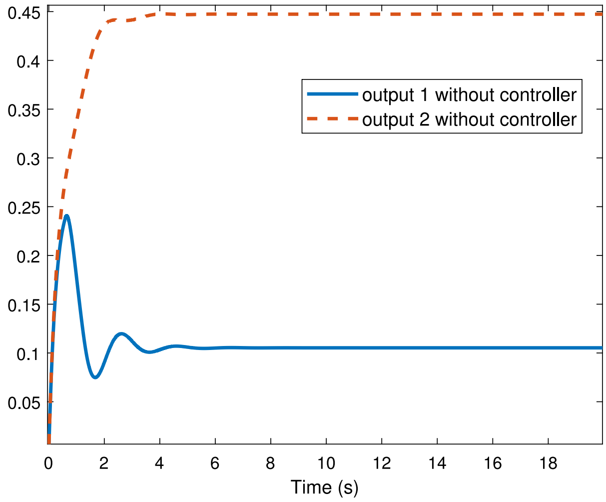

Fig. 15. Uncertain TDS without control for $\delta_1 = \delta_2 = -1$.

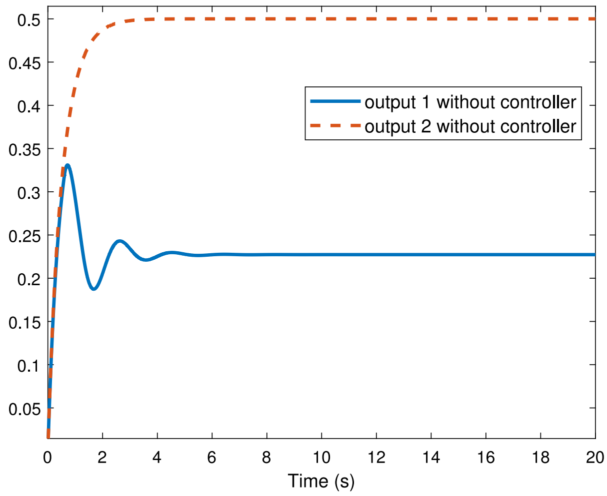

Fig. 16. Uncertain TDS without control for $\delta_1 = \delta_2 = 0$.

which means that robustness of performance is achieved. For simulation and replication of the results of this paper, MATLAB® and Simulink® version 2020a were used.

Example 5 (*The Cure of a Computationally Modelled Neuropathology [40,41]*)**.** In [40], the authors simulate a symptom of Alzheimer's disease (memory failure) using the Hopfield artificial neural network continuous model without delay. In that problem, a controller was designed to correct the effect of memory failure. A more realistic model would include time delay into the Eq. (8) of the model in [40], which could represent delay in neuron firing. The new time-delay model (adapted from Eq. (14) of [41]) is then:

$$\dot{u}_i(t) = -d_i u_i(t) + T_{i,n-(i-1)} \tanh[u_{n-(i-1)}(t)] + 0.24 F_{i,i} u_{n-(i-1)}(t-\tau) + I_i. \tag{34}$$

To have a model with 12 neurons, one uses $n = 12$ and $i = 1, 2, \ldots, 12$, the values d_i and $T = 0.5T_0$ are the same as for [40] and the F matrix is unit diagonal. The delay takes the value $\tau = 4.5$ seconds. In this example, in accordance with Figs. 19 and 20, the network cannot remember the letter L of 12 neurons ($I = [1, -1, -1, 1, -1, -1, 1, -1, -1, 1, 1, 1]$), the initial condition being $u^0 = [0, 0.5, 0, -1, 0, 0.5, 0, 1, 1, -1, 1, 1]$. Note that Fig. 19 shows the trajectories of the responses of the u_i-neurons over time, which are represented in Fig. 20 obtained using Matlab®'s function *imshow*(). Therefore, here one has a 12th order differential equation with a delay.

The controller is designed from a system with time delay and δ uncertainties instead of nonlinearities, which is valid for some classes of problems (Lurie problem). In this case, according to [26], the nonlinearities can be modelled as uncertainties, taking:

$$F_{(0,2]} := \{tanh(u_j) \,|\, \tanh(0) = 0, 0 < u_j tanh(u_j) \le 2u_j^2, u_j \neq 0\},$$

where $F_{(0,2]}$ is the set of all nonlinearities of Eq. (34) mapped in the first and third quadrants of the plane, with angular coefficients equal to 2. Thus, the fixed static nonlinearity $tanh(u)$ is replaced by a set of uncertain but fixed linear functions αu, where the

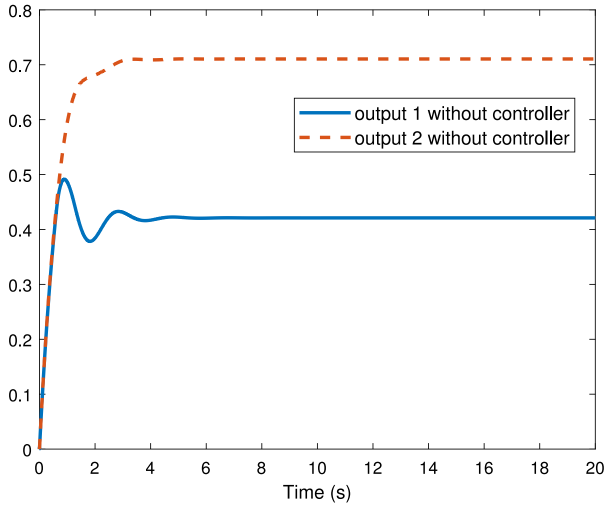

Fig. 17. Uncertain TDS without control $\delta_1 = \delta_2 = 1$.

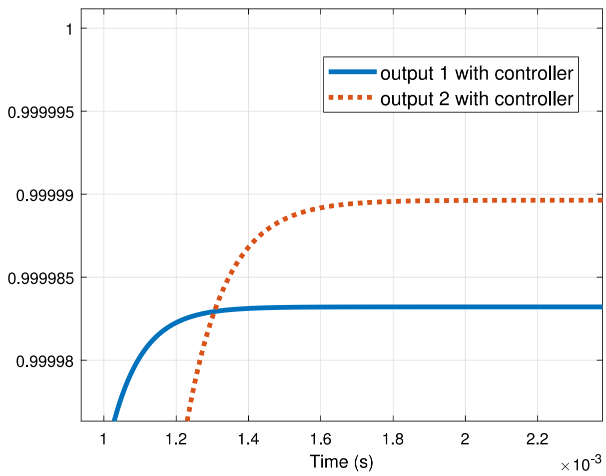

Fig. 18. Uncertain TDS in closed-loop control for $\delta_1 = \delta_2 = -1$, $\delta_1 = \delta_2 = 0$ and $\delta_1 = \delta_2 = 1$.

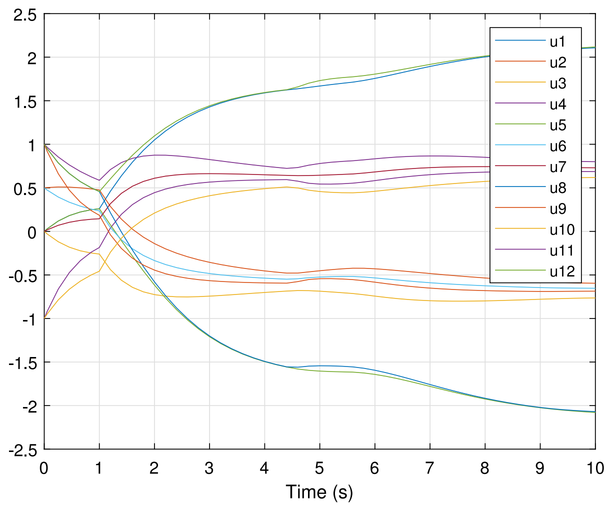

Fig. 19. Temporal response of the network with memory failure.

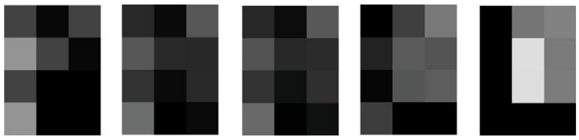

Fig. 20. The network cannot clearly remember the letter L.

parameters $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_{12}]$ are fixed but unknown real constants, such that, $0 < \alpha_j \leq 2$. Therefore, Eq. (34) takes on the new form:

$$\dot{u}_i(t) = -d_i u_i(t) + T_{i,n-(i-1)} \alpha u_{n-(i-1)}(t) + 0.24 F_{i,i} u_{n-(i-1)}(t-\tau) + I_i. \tag{35}$$

By carrying out the normalization process, one takes $\alpha_i = 1 + \delta_i$, $|\delta_i| \leq 1$, $i = 1, 2, \ldots, 12$. So, (35) becomes:

$$\dot{u}_i(t) = -d_i u_i(t) + T_{i,n-(i-1)} u_{n-(i-1)}(t) + T_{i,n-(i-1)} \delta_i + 0.24 F_{i,i} u_{n-(i-1)}(t-\tau) + I_i. \tag{36}$$

Taking the values $i = 1, \ldots, 12$ for Eq. (36), one obtains the explicit model of differential equations that simulate the memory fault:

$$\begin{cases} \dot{u}_1(t) & = & -u_1(t) & + & 0.75u_{12} & + & 0.75\delta_1 & + & 0.24u_{12}(t-\tau) & + & I_1, \\ \dot{u}_2(t) & = & -u_2(t) & + & 0.75u_{11} & + & 0.75\delta_2 & + & 0.24u_{11}(t-\tau) & + & I_2, \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ \dot{u}_{12}(t) & = & -u_{12}(t) & + & 0.75u_1 & + & 0.75\delta_{12} & + & 0.24u_1(t-\tau) & + & I_{12}. \end{cases} \tag{37}$$

Putting the system in the form of Eq. (1), one has:

$$A_0 = \begin{pmatrix} -1 & 0 & \cdots & 0 & 0.75 \\ 0 & -1 & \cdots & 0.75 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0.750 & \cdots & -1 & 0 \\ 0.75 & 0 & \cdots & 0 & -1 \end{pmatrix}_{[12\times12]}, \quad B = \begin{pmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{pmatrix}_{[12\times12]}, \quad C = \begin{pmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{pmatrix}_{[12\times12]},$$

$$F_\tau = \begin{pmatrix} 0.24 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0.24 \end{pmatrix}_{[12\times12]}, \quad \Delta A = \begin{pmatrix} 0 & \cdots & 0.75\delta_1 \\ \vdots & \iddots & \vdots \\ 0.75\delta_{12} & \cdots & 0 \end{pmatrix}_{[12\times12]} = \sum_{j=1}^{12} A_j \delta_j,$$

$$\text{where } A_1 = \begin{pmatrix} 0 & \cdots & 0.75 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{pmatrix}_{[12\times12]}, \quad A_2 = \begin{pmatrix} 0 & \cdots & 0 & 0 \\ 0 & \cdots & 0.75 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{pmatrix}_{[12\times12]}, \ldots, A_{12} = \begin{pmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0.75 & \cdots & 0 \end{pmatrix}_{[12\times12]}.$$

As $A_i = E_i G_i$ with $\rho_i = 1$, $E_1 = \begin{pmatrix} 0.75 \\ \vdots \\ 0 \end{pmatrix}_{[12\times1]}, \ldots, E_{12} = \begin{pmatrix} 0 \\ \vdots \\ 0.75 \end{pmatrix}_{[12\times1]}$ and

$$G_1 = \begin{pmatrix} 0 & \cdots & 1 \end{pmatrix}_{[1\times12]}, \ldots, G_{12} = \begin{pmatrix} 1 & \cdots & 0 \end{pmatrix}_{[1\times12]}.$$

Similar to the previous example, the DK-iteration and Theorem 9 are applied. After 3 iterations, as shown in Figs. 21 and 22, the cure for the memory fault is obtained, and the network remembers the letter.

6. Conclusion and future work

In this paper, new methodologies for analysis and design of uncertain linear time-invariant time-delay systems (ULTITDS) were presented. The techniques are extensions, for time-delay systems, of necessary and sufficient condition for robustness of stability and performance, as well as design methodologies. Several auxiliary original results were presented (in order to prove the main theorems) that are interesting in its own right. Padé approximation theory, as well as H_∞ robust control results, were used in the development of the presented methods. In particular, the existence of an upper bound function $m_{delay}(j\omega)$ for $\mu_{delay}(j\omega)$, that is, the structured singular value for a linear time-invariant time-delay system, was proved. This function is then used in the analysis and design methodology in order to verify and synthesize robust systems. As far as the authors know, such results are not in the literature.

Theorem 4, which is the main theorem proved in the paper, is a generalization of one from Skogestad [20], and was presented, in a version for non-linear systems, as a conjecture in Pinheiro and Colón [26,35]. To the best of the authors' knowledge, the results

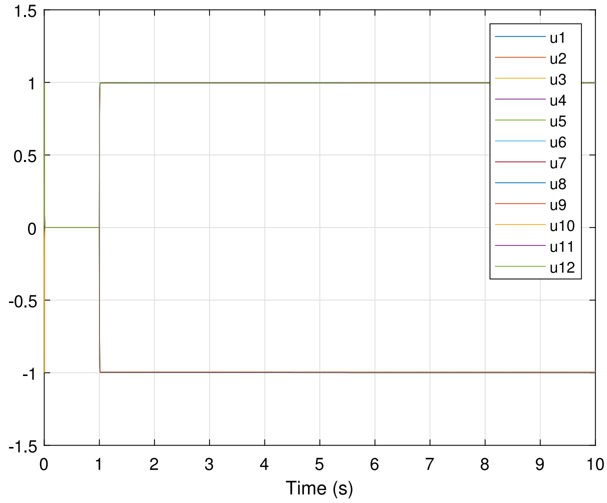

Fig. 21. Temporal response of the network with control.

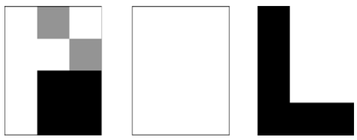

Fig. 22. In this case, the network remembers the letter (cured).

from section four till the end are original contributions. Examples were presented which confirm the theoretical results obtained and show that those methods are less conservative than others in literature. For future research, the following can be highlighted:

- One can extend the results for ULTITDS with two or more different time-delay values.
- One can extend the results for ULTITDS with uncertainties multiplying the time-delay terms.
- Extend the theory to the discrete time domain.
- Extend the theory to uncertain time-delays.
- Estimation of the velocity of convergence of the Padé approximation will be pursued.

In the field of applications, the theory presented in this paper can be applied to various problems involving ULTITDS (see [2,5,12]). Some specific possibilities are:

- Renewable energy industry, where the results can be useful in the design of robust controllers for power take-off (PTO) of wave energy converters (WEC) with delays. It is observed in works such as [59–64] the need of robust controllers, as in WEC-PTO systems there are many non-linearities (which can be modelled as uncertainties). In the paper [64] the author asserts that robust control of the WEC-PTO is a significant emerging area as it aims to insert uncertainties into WEC models in order to find robust control and greater robustness in wave energy extraction.
- Neuroscience, more specifically, in modelling and control of neuropathologies [40,41]. The idea is to obtain controllers that could be implemented in neural implants to relieve the effects of the Alzheimer disease.
- Epidemiological mathematical models, like the one presented in [65–67], which establishes an epidemiological model of COVID-19 with five variables. Uncertainties in the parameters are common in such models, as well as time-delays, and control techniques could be devised in order to control the spread of the disease.
- Cardiac devices [68–70], since there are many variables with uncertainties and time-delays in such cases. For example, in [69] the authors deal with automatic heart rate control in cycle ergometer exercise. This is an important component, especially for patients who are in the process of rehabilitation, to ensure that their limits are not exceeded and that the exercise does not cause damage to their health. Other possibility of application could be in controlling pressure in diabetic foot by using insole of new biomaterials [71]. Such devices have significant uncertainties and delay and can be enhanced by means of pressure actuators in a closed loop scheme.
- Finally, some techniques for linear time-invariant systems can be generalized for systems described using Lie groups and Differential Geometry [72–74]

Ethical statement

There are not human and animal subject in this article, and informed consent is not applicable.

CRediT authorship contribution statement

Rafael Fernandes Pinheiro: Conceptualization, Formal analysis, Investigation, Methodology, Software, Validation, Writing – original draft, Writing – review & editing. **Diego Colón:** Conceptualization, Formal analysis, Investigation, Methodology, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] L. Wu, H.K. Lam, Y. Zhao, Z. Shu, Time-delay systems and their applications in engineering, Math. Probl. Eng. 2015 (2015) 246351.
[2] J.P. Richard, Time-delay systems: an overview of some recent advances and open problems, Automatica 39 (10) (2003) 1667–1694.
[3] B. Song, J.H. Park, Z.G. Wu, X. Li, New results on delay-dependent stability analysis and stabilization for stochastic time-delay systems, Internat. J. Robust Nonlinear Control 24 (16) (2014) 2546–2559.
[4] E. Fridman, U. Shared, Special issue on time-delay systems, Internat. J. Robust Nonlinear Control 13 (9) (2003) 791–792.
[5] S. Mondié, A. Egorov, Special issue on system theory and delay in honor of Vladimir Kharitonov, Internat. J. Robust Nonlinear Control 32 (6) (2022) 3099–3100.
[6] Z.-G. Liu, L. Xue, W. Zhang, Universal adaptive control strategies for stochastic nonlinear time-delay systems with odd rational powers, Automatica 125 (2021) 109419.
[7] J. Hale, Theory of Functional Differential Equations, in: Applied Mathematical Sciences, Springer New York, 1977.
[8] E. Fridman, Introduction to Time-Delay Systems: Analysis and Control, Springer International Publishing, Switzerland, 2014.
[9] K. Gu, J. Chen, V.L. Kharitonov, Stability of Time-Delay Systems, Springer Science & Business Media, 2003.
[10] N. Luo, M. De, L.A. Sen, J. Rodellar, Robust stabilization of a class of uncertain time delay systems in sliding mode, Internat. J. Robust Nonlinear Control 7 (1997) 59–74.
[11] Y.P. Huang, K. Zhou, Robust stability of uncertain time-delay systems, IEEE Trans. Automat. Control 45 (11) (2000) 2169–2173.
[12] R. Xu, Z. Liu, Y. Liu, State-estimation-based adaptive sliding mode control for a class of uncertain time-delay systems: a new design, Internat. J. Systems Sci. 53 (2) (2021) 375–387.
[13] J. Akiba, K.-Z. Liu, L. Qiu, P. Yu, K. Koiwa, T. Zanma, A less conservative robust control method for a class of nonpassive uncertain systems, Internat. J. Robust Nonlinear Control 32 (2) (2022) 682–697.
[14] Y. Wang, C. Hua, Y. Qiu, Robust stability and H_∞ control for networked control systems with transmission delay and its application to 2 DoF laboratory helicopter, J. Franklin Inst. B 360 (4) (2023) 2827–2847.
[15] D. Bresch-Pietri, M. Krstic, Delay-adaptive control for nonlinear systems, IEEE Trans. Automat. Control 59 (5) (2014) 1203–1218.
[16] C. Hua, L. Zhang, X. Guan, Robust Control for Nonlinear Time-Delay Systems, Springer, 2018.
[17] R. Pongvuthithum, K. Rattanamongkhonkun, W. Lin, Asymptotic regulation of time-delay nonlinear systems with unknown control directions, IEEE Trans. Automat. Control 63 (5) (2017) 1495–1502.
[18] Z.-G. Liu, L. Xue, Z.-Y. Sun, A new robust adaptive tracking strategy to uncertain time-delay nonlinear systems with a general form, Automatica 146 (2022) 110560.
[19] J. Doyle, Analysis of feedback systems with structured uncertainties, IEE Proc. D: Control Theory Appl. 129 (1982) 242–250.
[20] S. Skogestad, I. Postlethwaite, Multivariable Feedback Control: Analysis and Design, second ed., John Wiley and Sons, New York, 2007.
[21] G. Baker, P.R. Grave-Morris, Padé Approximants, Addison-Wesley Publishing Company, Massachusetts, 1981.
[22] T. Insperger, On the approximation of delayed systems by taylor series expansion, J. Comput. Nonlinear Dyn. 10 (2) (2015).
[23] Y. Wei, Y. Hu, Y. Dai, Y. Wang, A generalized padé approximation of time delay operator, Int. J. Control Autom. Syst. 14 (1) (2016) 181–187.
[24] J.R. Partington, Approximation of delay systems by Fourier-Laguerre series, Automatica 27 (3) (1991) 569–572.
[25] M.G. Yoon, B.H. Lee, A new approximation method for time-delay systems, IEEE Trans. Automat. Control 42 (7) (1997) 1008–1012.
[26] R.F. Pinheiro, D. Colón, On the μ-analysis and synthesis of MIMO Lurie-type systems with application in complex networks, Circuits Systems Signal Process. 40 (1) (2021) 193–232.
[27] R.F. Pinheiro, D. Colón, An application of the lurie problem in hopfield neural networks, in: Proceedings of DINAME 2017: Selected Papers of the XVII International Symposium on Dynamic Problems of Mechanics 17, Springer, 2019, pp. 371–382.
[28] F. Goktas, J.M. Smith, R. Bajcsy, μ-Synthesis for distributed control systems with network-induced delays, in: Proceedings of the IEEE Conference on Decision and Control, Vol. 1, 1996.
[29] Z. Feng, Y. Yang, Robust H_∞ sliding mode control for uncertain polynomial fuzzy stochastic singular systems, J. Franklin Inst. B 360 (4) (2023) 2893–2912.
[30] H. Yi, Z. gang Su, P. hong Wang, Robust tuning of uncertainty and disturbance estimator-based control for stable processes with time delay, J. Franklin Inst. B 360 (4) (2023) 2594–2616.
[31] Z.Q. Wang, S. Skogestad, μ-Analysis and synthesis of time delay systems using smith predictor, in: Proceedings of the IEEE Conference on Decision and Control, Vol. 3, 1991, pp. 3033–3034.
[32] L. Chen, C. Tang, Z. Liu, Study on robust control for engine idle speed based on structured singular value μ-theory, in: Proceedings - 3rd International Conference on Measuring Technology and Mechatronics Automation, Vol. 3, 2011, pp. 514–517.
[33] L. Pavel, A μ-analysis application to stability of optical networks, in: Proceedings of the 2004 American Control Conference, Vol. 5, 2004, pp. 3956–3961.
[34] R.F. Pinheiro, The Lurie Problem and Its Relationships with Artificial Neural Networks and Alzheimer-Like Disease (Ph.D. thesis), University of São Paulo, São Paulo, 2021..
[35] R.F. Pinheiro, D. Colón, Analysis and synthesis of single-input-single-output Lurie type systems via $\mathcal{H}_\infty$ mixed-sensitivity, Trans. Inst. Meas. Control 44 (1) (2022) 133–143.
[36] A.I. Lurie, V.N. Postnikov, On the theory of stability of control systems, Prikl. Mat. Mekh. 8 (3) (1944) 246–248 (in Russian).
[37] M. Carter, Foundations of Mathematical Economics, MIT Press, 2001.
[38] D.-W. Gu, P. Petkov, M.M. Konstantinov, Robust Control Design with MATLAB®, Springer Science & Business Media, 2005.
[39] Z. Li, F. Khasawneh, X. Yin, A. Li, Z. Song, A new microscopic traffic model using a spring-mass-damper-clutch system, IEEE Trans. Intell. Transp. Syst. 21 (8) (2020) 3322–3331.
[40] R.F. Pinheiro, D. Colón, L.L.R. Reinoso, Relating Lurie's problem, Hopfield's network and Alzheimer's disease, Congresso Brasileiro Automática - CBA 2 (1) (2020).

[41] R.F. Pinheiro, D. Colón, R. Fonseca-Pinto, An improved Alzheimer-like disease computational model via delayed hopfield network with Lurie control system for healing, 2023, TechRxiv. Preprint.
[42] D. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, in: Athena scientific optimization and computation series, Athena Scientific, 1996.
[43] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.
[44] E.A. Feinberg, P.O. Kasyanov, M. Voorneveld, Berge's maximum theorem for noncompact image sets, J. Math. Anal. Appl. 413 (2) (2014) 1040–1046.
[45] G. Baker, P. Graves-Morris, Padé Approximants: Basic Theory, in: Encyclopedia of Mathematics and Its Applications, vol. 1, Addison-Wesley, Advanced Book Program, 1981.
[46] J. Lam, Model reduction of delay systems using Pade approximants, Internat. J. Control 57 (2) (1993) 377–391.
[47] E. Freitag, R. Busam, Complex Analysis, in: Universitext (1979), Springer, 2005.
[48] E.B. Saff, R.S. Varga, On the zeros and poles of Padè approximants to e^z, Numer. Math. 25 (1) (1975) 1–14.
[49] E.B. Saff, R.S. Varga, On the zeros and poles of Padè approximants to e^z III, Numer. Math. 26 (2) (1976) 241–266.
[50] B.G. Morton, R.M. McAfoos, A mu-test for robustness analysis of a real-parameter variation problem, in: 1985 American Control Conference, IEEE, 1985, pp. 135–138.
[51] K. Zhou, J.C. Doyle, K. Glover, Optimal Control, Prentice Hall, New Jersey, 1995.
[52] W. Rudin, Real and Complex Analysis, in: Mathematics series, McGraw-Hill, 1987.
[53] P. Lancaster, M. Tismenetsky, The Theory of Matrices: With Applications, Academic Press, San Diego, CA, 1985.
[54] C.D. Meyer, Matrix Analysis and Applied Linear Algebra, SIAM, Philadelphia, 2000.
[55] E. Fridman, U. Shaked, An improved stabilization method for linear time-delay systems, IEEE Trans. Autom. Control 47 (11) (2002) 1931–1937.
[56] E. Fridman, U. Shaked, Parameter dependent stability and stabilization of uncertain time-delay systems, IEEE Trans. Automat. Control 48 (5) (2003) 861–866.
[57] C.R. Knospe, M. Roozbehani, Stability of linear systems with interval time delays excluding zero, IEEE Trans. Automat. Control 51 (8) (2006) 1271–1288.
[58] F. Lescher, C. Roos, Robust stability of time-delay systems with structured uncertainties: A μ-analysis based algorithm, in: 2011 50th IEEE Conference on Decision and Control and European Control Conference, 2011, pp. 4955–4960.
[59] J. Gaspar, R.F. Pinheiro, M. Kamarlouei, C. Guedes Soares, M.J.G.C. Mendes, Review on hardware-in-the-loop simulation of wave energy converters, in: Trends in Maritime Technology and Engineering, vol. 1, Taylor & Francis Group, 2022.
[60] J.F. Gaspar, P.K. Stansby, M. Calvário, C. Guedes Soares, Hydraulic power take-off concept for the M4 wave energy converter, Appl. Ocean Res. 106 (2021) 102462.
[61] J.F. Gaspar, M. Kamarlouei, A. Sinha, H. Xu, M. Calvário, F.X. Faÿ, E. Robles, C.G. Soares, Speed control of oil-hydraulic power take-off system for oscillating body type wave energy converters, Renew. Energy 97 (2016) 769–783.
[62] Y. Lao, J.T. Scruggs, Robust control of wave energy converters using unstructured uncertainty, in: 2020 American Control Conference, ACC, IEEE, 2020, pp. 4237–4244.
[63] J.V. Ringwood, A. Merigaud, N. Faedo, F. Fusco, Wave energy control systems: robustness issues, IFAC-PapersOnLine 51 (29) (2018) 62–67.
[64] J.V. Ringwood, Wave energy control: status and perspectives 2020, IFAC-PapersOnLine 53 (2) (2020) 12271–12282.
[65] C.J. Silva, C. Cruz, D.F. Torres, A.P. Muñuzuri, A. Carballosa, I. Area, J.J. Nieto, R. Fonseca-Pinto, R. Passadouro, E.S. dos Santos, W. Abreu, J. Mira, Optimal control of the COVID-19 pandemic: controlled sanitary deconfinement in Portugal, Sci. Rep. 11 (1) (2021) 1–15, arXiv:2009.00660.
[66] C.J. Silva, G. Cantin, C. Cruz, R. Fonseca-Pinto, R. Passadouro, E. Soares dos Santos, D.F. Torres, Complex network model for COVID-19: Human behavior, pseudo-periodic solutions and multiple epidemic waves, J. Math. Anal. Appl. 514 (2) (2022) 125171, arXiv:2010.02368.
[67] B. Pavlack, M. Grave, Dantas, et al., EPIDEMIC: Epidemiology educational code, J. Open Source Educ. 5 (50) (2022) 149.
[68] I. Birs, D. Copot, C.I. Muresan, et al., Robust fractional order PI control for Cardiac output stabilisation, IFAC-PapersOnLine 52 (1) (2019) 994–999.
[69] K.J. Hunt, C.C. Hurni, Robust control of heart rate for cycle ergometer exercise, Med. Biol. Eng. Comput. 57 (11) (2019) 2471–2482.
[70] M. Ali Akbar Semnani, A.R. Vali, S.M. Hakimi, V. Behnamgol, Modelling and design of observer based smooth sliding mode controller for heart rhythm regulation, Int. J. Dyn. Control 10 (3) (2022) 828–842.
[71] S. de S. R. F. Rosa, M. do C. Reis, M.F.F. Rosa, D. Cólon, C.A. dos Reis, C.A. dos Reis, J.M. Balthazar, Use of natural latex as a biomaterial for the treatment of diabetic foot — A new approach to treating symptoms of diabetes mellitus, in: D. Claborn (Ed.), Topics in Public Health, Intech, 2015, pp. 213–248.
[72] F. Pait, D. Colón, Some properties of the Riemannian distance function and the position vector X, with applications to the construction of Lyapunov functions, in: 49th IEEE Conference on Decision and Control, CDC, 2010, 2010, pp. 6277–6280.
[73] D. Colón, Connections between screw theory and cartan's connections, in: Proceedings of the Congresso Brasileiro de Automática 2014, Sociedade Brasileira de Automatica, Belo Horizonte, 2014, pp. 936–943.
[74] D. Colón, F.M. Pait, Geometry of adaptive control: Optimization and geodesics, Internat. J. Adapt. Control Signal Process. 18 (2004) 381–392.