Design of robust structurally constrained controllers for MIMO plants with time-delays

Deesh Dileep[1], Wim Michiels[2], Laurentiu Hetel[3], and Jean-Pierre Richard[4]

Abstract— The structurally constrained controller design problem for linear time invariant neutral and retarded time-delay systems (TDS) is considered in this paper. The closed-loop system of the plant and structurally constrained controller is modelled by a system of delay differential algebraic equations (DDAEs). A robust controller design approach using the existing spectrum based stabilisation and the H-infinity norm optimisation of DDAEs has been proposed. A MATLAB based tool has been made available to realise this approach. This tool allows the designer to select the sub-controller input-output interactions and fix their orders. The results obtained while stabilising and optimising two TDS using structurally constrained (decentralised and overlapping) controllers have been presented in this paper.

Index Terms— Decentralized control, Time-delay systems, H2/H-infinity methods, linear systems, Large-scale systems.

I. INTRODUCTION

This article contributes to the field of complex interconnected dynamical systems with time-delays. It is common to observe time-delays in these systems due to their inherent properties or due to the delays in communication. It is almost infeasible, if not, costly to implement centralised controllers for large scale dynamical systems (see [1] and references within). Therefore, decentralised or overlapping controllers are often considered as favourable alternatives.

There are many methods suggested by multiple authors for the design of full order controllers that stabilise finite dimensional LTI MIMO systems. The design problem of such a controller is usually translated into a convex optimisation problem expressed in terms of linear matrix inequalities (LMIs). However, determining a reduced dimension (order) controller or imposing special structural constrains on the controller introduces complexity. Since the constraints on structure or dimension prevent a formulation in terms of LMIs. Such problems typically lead to solving bilinear matrix inequalities directly or using other non-convex optimisation techniques. Solutions obtaining full order controllers for higher order plants are not favourable, since lower order controllers are preferred for implementation.

Time-delay systems (TDS) can be seen as infinite dimensional LTI MIMO systems. Designing a finite dimensional

controller for TDS is hence equivalent to obtaining a reduced order controller. Therefore, in this paper we combine both the problems of determining a reduced order (or fixed structure) controller and imposing constrains on the structure of the controller.

Linear time invariant (LTI) neutral (and retarded) time-delay systems are considered in this article. The algorithms from [2] and [3] with a direct optimisation based approach have been extended in this paper for designing structurally constrained robust controllers. In [3], the design of stabilising fixed-order controllers for TDS has been translated into solving a non-smooth non-convex optimisation problem of minimising the spectral abscissa. This approach is similar in concept to the design of reduced-order controllers for LTI systems as implemented in the HIFOO package (see [4]). The core algorithm of HANSO matlab code is used for solving the non-smooth non-convex optimisation problems (see [5]). In many control applications, robust design requirements are usually defined in terms of $\mathcal{H}_\infty$ norms of the closed-loop transfer function including the plant, the controller, and weights for uncertainties and disturbances. In [2], the design of a robust fixed-order controller for TDS has been translated into a non-smooth non-convex optimisation problem. There are other methods available to design optimal $\mathcal{H}_\infty$ controllers for LTI finite dimensional MIMO systems based on Riccati equations and linear matrix inequalities (see [6], [7], and references within). However, the order of the controller designed by these methods is generally larger than or equal to the order of the plant. Also, imposing structural constrains in these controllers become difficult.

There are many methods available to design decentralised controllers for non-delay systems, most of them do not carry over easily to the case of systems with time-delays. In this paper, the direct optimisation problem of designing overlapping or decentralised controllers is dealt with by imposing constrains on the controller parameters. Similar structural constrain methodologies were already mentioned in [1], [8], [9], and [10].

This work allows system models in terms of delay-differential algebraic equations (DDAEs), whose power in modelling large classes of delay equations is illustrated in the next section. In [2], the authors state that such a system description form can be adapted for designing controllers due to the generality in modelling interconnected systems and controllers. In this way, elimination technique can be avoided which might not be possible for systems with delays. In the DDAE form, the linearity of the closed-loop system, with respect to the matrices of the controllers, can be preserved

[1]Department of Computer Science, KU Leuven, 3001 Leuven, Belgium `deesh.dileep@cs.kuleuven.be`

[2]Department of Computer Science, KU Leuven, 3001 Leuven, Belgium `wim.michiels@cs.kuleuven.be`

[3]Centrale Lille, CNRS UMR 8179 CRIStAL, Lille, France. `laurentiu.hetel@centralelille.fr`

[4]Centrale Lille, Inria project-team Non-A, CNRS UMR 8179 CRIStAL, Lille, France. `jean-pierre.richard@centralelille.fr`

for various types of delays and combinations of plants and controllers.

The rest of the paper is organised as follows. Section II formally introduces the problem of time-delay systems and the existing methods available to stabilise and optimise the performance of such systems using centralised fixed-order controllers. Section III presents the proposed concept of structurally constrained controllers and its implementation methodology. Section IV provides some example MIMO problems from literature which are stabilised and optimised using structurally constrained controllers. Section V concludes the paper with a few remarks.

II. PRELIMINARIES

In this article, TDS or plants of the following form are considered,

$$E_p \dot{x}_p(t) = A_{p0}x_p(t) + \sum_{i=1}^{m^A} A_{pi}x_p(t - h_i^A)$$
$$+ B_{p1}u(t) + B_{p2}w(t), \quad (1)$$
$$y(t) = C_{p1}x_p(t),$$
$$z(t) = C_{p2}x_p(t).$$

Here t is the time variable, $x_p(t) \in \mathbb{R}^n$ is the instantaneous state vector at time t, similarly, $u(t) \in \mathbb{R}^w$ and $y(t) \in \mathbb{R}^z$ are instantaneous controlled input and measured output vectors respectively at time t. We use the notations $\mathbb{R}$, $\mathbb{R}^+$ and $\mathbb{R}_0^+$ to represent sets of real numbers, non-negative real numbers and strictly positive real numbers respectively, and $x_p \in \mathbb{R}^n$ is a short notation for $(x_{p1}, ..., x_{pn})$. A, B, C, D and E are constant real-valued matrices, m^A is a positive integer representing the number of distinct time-delays present in the state, the inputs, the outputs, the feed-through (input-output) and the first order derivative of instantaneous state vector. The time-delays, $0 < h_i^A \leq h_{max}$, have a minimum value greater than zero and a maximum value of h_{max}. The instantaneous exogenous input and the instantaneous exogenous (or controlled) output are represented as $w(t)$ and $z(t)$ respectively.

Even though there are no feed-through components, input delays or output delays, the LTI system description of (1) is in the most general form. This can be portrayed with the help of some simple examples.

Example 1. Consider a system with non-trivial feed-through matrices.

$$\begin{cases} \dot{\psi}(t) &= A\psi(t) + B_1 u(t) + B_2 w(t) \\ y(t) &= C_1\psi(t) + D_{11}u(t) + D_{12}w(t) \\ z(t) &= C_2\psi(t) + D_{21}u(t) + D_{22}w(t) \end{cases}$$

If we consider $x_p(t) = [\psi(t)^T \gamma_u(t)^T \gamma_w(t)^T]^T$, we can bring this system to the form of (1) with the help of the dummy variables (γ_u and γ_w),

$$\begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\dot{x}_p(t) = \begin{bmatrix} A & B_1 & B_2 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} x_p(t) + \begin{bmatrix} 0 \\ -I \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ 0 \\ -I \end{bmatrix} w(t),$$
$$y(t) = \begin{bmatrix} C_1 & D_{11} & D_{12} \end{bmatrix} x_p(t), \quad z(t) = \begin{bmatrix} C_2 & D_{21} & D_{22} \end{bmatrix} x(t).$$

Example 2. Consider an LTI system with time-delays at the input.

$$\begin{cases} \dot{\psi}(t) = A\psi(t) + B_{10}u(t) + \sum_{i=1}^{m^B} B_{1i}u(t - h_i^B) \\ y(t) = C_1\psi(t) + D_{11}u(t) \end{cases}$$

If we consider $x_p(t) = [\psi(t)^T \gamma_u(t)^T]^T$, we can bring this system to the form of (1) with the help of the dummy variable (γ_u),

$$\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}\dot{x}_p(t) = \begin{bmatrix} A & B_{10} \\ 0 & I \end{bmatrix} x_p(t)$$
$$+ \sum_{i=1}^{m^B} \begin{bmatrix} 0 & B_{1i} \\ 0 & 0 \end{bmatrix} x_p(t - h_i^B) + \begin{bmatrix} 0 \\ -I \end{bmatrix} u(t),$$
$$y(t) = \begin{bmatrix} C_1 & D_{11} \end{bmatrix} x_p(t).$$

Simliarly, the output delays can be virtually "eliminated".

Example 3. The presence of time-delays at the first order derivative of the state vector in an LTI system (neutral equation) can also be virtually eliminated using dummy variables.

$$\begin{cases} \dot{\psi}(t) + \sum_{i=1}^{m^E} E_i\dot{\psi}(t - h_i^E) &= A\psi(t) + B_1 u(t) \\ y(t) &= C_1\psi(t) + D_{11}u(t) \end{cases}$$

We can bring this example LTI system to the form of (1) with the help of the dummy variables (γ_ψ and γ_u), where γ_ψ is given by,

$$\gamma_\psi(t) = \psi(t) + \sum_{i=1}^{m^E} E_i\psi(t - h_i^E).$$

More precisely, when defining $x_p(t) = [\gamma_\psi(t)^T \psi(t)^T \gamma_u(t)^T]^T$ the system takes the following form consistent with (1):

$$\begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\dot{x}_p(t) = \begin{bmatrix} 0 & A & B_1 \\ -I & I & 0 \\ 0 & 0 & I \end{bmatrix} x_p(t)$$
$$+ \sum_{i=1}^{m^E} \begin{bmatrix} 0 & 0 & 0 \\ 0 & E_i & 0 \\ 0 & 0 & 0 \end{bmatrix} x_p(t - h_i^E) + \begin{bmatrix} 0 \\ 0 \\ -I \end{bmatrix} u(t)$$
$$y(t) = \begin{bmatrix} 0 & C_1 & D_{11} \end{bmatrix} x_p(t).$$

The system described in (1) could be controlled using the following feedback controller of the prescribed order "n_c",

$$\dot{x}_c(t) = A_c x_c(t) + B_c y(t),$$
$$u(t) = C_c x_c(t) + D_c y(t). \quad (2)$$

The case of $n_c = 0$ corresponds to a static or proportional controller of the form $u(t) = D_c y(t)$. The other cases of $n_c \geq 1$ corresponds to that of a dynamic controller as in the form (2), where, A_c is a matrix of size $n_c \times n_c$.

The combination of the plant (1) and the feedback controller (2) can be re-written using

$$x = [x_p^T u^T \gamma_w^T x_c^T y^T]^T, \quad (3)$$

in the general form of delay differential algebraic equation (DDAE) as shown below,

$$E\dot{x}(t) = A_0 x(t) + \sum_{i=1}^{m} A_i x(t - \tau_i) + Bw(t),$$
$$z(t) = Cx(t),$$

(4)

where,

$$E = \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$A_0 = \begin{bmatrix} A_{p0} & B_{p1} & B_{p2} & 0 & 0 \\ C_{p1} & 0 & 0 & 0 & -I \\ 0 & 0 & -I & 0 & 0 \\ 0 & 0 & 0 & \bar{A}_c & \bar{B}_c \\ 0 & -I & 0 & \bar{C}_c & \bar{D}_c \end{bmatrix}.$$

(5)

Subsequently,

$$A_i = \begin{bmatrix} A_{pi} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ 0 \\ I \\ 0 \\ 0 \end{bmatrix}, \quad C^{\mathrm{T}} = \begin{bmatrix} C_{p2} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

An useful property of this modelling approach using DDAEs is the linear dependence of closed-loop system matrices on the elements of the controller matrices. To stabilise and optimise the robustness of the closed-loop system, the time-independent parameter vector of $\bar{p}$ is defined. We build on the approach of [3] and [2], that is to directly optimise stability and performance measures as a function of vector $\bar{p}$, which contains the parameters of the controller,

$$\bar{p} = \mathrm{vec}\left(\begin{bmatrix} A_c & B_c \\ C_c & D_c \end{bmatrix} \right).$$

(6)

For a centralised controller, the matrices A_c, B_c, C_c and D_c are seldom sparse when computed using the algorithms presented in [3] or [2]. Static controllers can be considered as a special case of the dynamic controller, for which A_c, B_c and C_c are empty matrices. The vector $\bar{p}$ would then include only the elements from D_c.

The objective functions used for the performance evaluation of the closed-loop system will be explained in the following subsections.

A. Robust Spectral Abscissa optimisation:

The spectral abscissa ($c(\bar{p})$) of the closed-loop system (4) when $w \equiv 0$ can be expressed as follows,

$$c(\bar{p}; \bar{\tau}) = \sup_{\lambda \in \mathbb{C}} \{ \Re(\lambda) : \det\Delta(\lambda, \bar{p}; \bar{\tau}) = 0 \},$$

where,

(7)

$$\Delta(\lambda, \bar{p}; \bar{\tau}) = \lambda E - A_0(\bar{p}) - \sum_{i=1}^{m} A_i(\bar{p}) e^{-\lambda \tau_i}$$

and $\Re(\lambda)$ is the real part of the complex number λ. The exponential stability of the null solution of (4) determined by the condition $c(\bar{p}) < 0$ (see [3]). However, the function $\bar{\tau} \in (\mathbb{R}_0^+)^m \mapsto c(\bar{p}; \bar{\tau})$ might not be continuous and could be sensitive to infinitesimal delay changes (in general, as neutral TDS could be included in (1)). Therefore, we define the robust spectral abscissa $C(\bar{p}; \bar{\tau})$ as in the following way,

$$C(\bar{p}; \bar{\tau}) := \lim_{\epsilon \to 0+} \sup_{\bar{\tau}_e \in \mathcal{B}(\bar{\tau}, \epsilon)} c(\bar{p}; \bar{\tau}_\epsilon)$$

(8)

In (8), $\mathcal{B}(\bar{\tau}, \epsilon)$ is an open ball of radius $\epsilon \in \mathbb{R}^+$ centered at $\bar{\tau} \in (\mathbb{R}^+)^m$, $\mathcal{B}(\bar{\tau}, \epsilon) := \{ \bar{\theta} \in \mathbb{R}^m : ||\bar{\theta} - \bar{\tau}|| < \epsilon \}$. The sensitivity of the spectral abscissa with respect to infinitesimal delay perturbations has been resolved by considering the robust spectral abscissa, since this function can be shown to be a continuous function of the delay parameters (and also parameters in $\bar{p}$), see [3]. We now define the concept of strong exponential stability.

Definition 1: The null solution of (4) when $w \equiv 0$ is strongly exponentially stable if there exists a number $\hat{\tau} > 0$ such that the null solution of

$$E\dot{x}(t) = A_0 x(t) + \sum_{i=1}^{m} A_i x(t - (\tau_i + \delta\tau_i)))$$

is exponentially stable for all $\delta\bar{\tau} \in \mathbb{R}^m$ satisfying $||\delta\bar{\tau}|| < \hat{\tau}$ and $\tau_i + \delta\tau_i \geq 0$, $i = 1, ..., m$.

In [3] it has been shown that the null solution is strongly exponentially stable iff $C(\bar{p}) < 0$. To obtain a strongly exponentially stable closed-loop system and to maximise the exponential decay rate of the solutions, the controller parameters (in $\bar{p}$) are optimised for minimum robust spectral abscissa, that is,

$$\min_{\bar{p}} \longrightarrow C(\bar{p}).$$

(9)

B. Strong $\mathcal{H}_\infty$ norm optimisation

The transfer function from w to z of the system represented by (4) is given by,

$$G_{zw}(\lambda, \bar{p}; \bar{\tau}) := C\left(\lambda E - A_0(\bar{p}) - \sum_{i=1}^{m} A_i(\bar{p}) e^{-\lambda \tau_i} \right)^{-1} B.$$

(10)

The $\mathcal{H}_\infty$ norm for a stable system with the transfer function given in (10) can be expressed as,

$$||G_{zw}(j\omega, \bar{p}; \bar{\tau})||_\infty = \sup_{\omega \in \mathbb{R}} \sigma_1(G_{zw}(j\omega, \bar{p}; \bar{\tau})).$$

(11)

However, similar to the spectral abscissa, the function $\bar{\tau} \in (\mathbb{R}_0^+)^m \mapsto ||G_{zw}(j\omega, \bar{p}; \bar{\tau})||_\infty$ might not be continuous and

could be sensitive to infinitesimal delay changes (in general, inherent from the behaviour of the transfer function at high frequencies). Therefore, under the assumption of strong exponential stability of the null solution, we define the strong $\mathcal{H}_\infty$ norm $|||G_{zw}(j\omega, \bar{p}; \bar{\tau})|||_\infty$.

$$|||G_{zw}(j\omega, \bar{p}; \bar{\tau})|||_\infty := \lim_{\epsilon \to 0+} \sup_{\bar{\tau}_e \in \mathcal{B}(\bar{\tau}, \epsilon)} ||G_{zw}(j\omega, \bar{p}; \bar{\tau}_e)||_\infty$$

Contrary to the (standard) $\mathcal{H}_\infty$ norm, the strong $\mathcal{H}_\infty$ norm continuously depends on the delay parameter. The continuous dependence also holds with respect to the elements of the system matrices, which includes the elements in $\bar{p}$ (see [2]).

To improve the robustness expressed in terms of a $\mathcal{H}_\infty$ criterion, controller parameters (in $\bar{p}$) are optimised for minimum strong $\mathcal{H}_\infty$ norm. This brings us to the optimisation problem,

$$\min_{\bar{p}} \longrightarrow |||G_{zw}(j\omega, \bar{p}; \bar{\tau})|||_\infty.$$

To solve the non-smooth non-convex objective function involving the strong $\mathcal{H}_\infty$ norm, it is essential to start with an initial set of controller parameters for which the closed-loop system is strongly exponentially stable. If this is not the case, a preliminary optimisation is performed based on minimising the robust spectral abscissa.

Weighted sum approach: A simple weighted sum based optimisation approach can also be performed using the two objectives mentioned in the sub-sections II-A and II-B. The controller parameters (in $\bar{p}$) can be optimised for the minimum of a multi-objective function $f_o(\bar{p})$, that is,

$$\min_{\bar{p}} f_o(\bar{p}), \qquad (12)$$

where,

$$f_o(\bar{p}) \begin{cases} \infty, & \text{if} \quad C(\bar{p}) \geq 0 \\ \alpha \; C(\bar{p}) + \; (1-\alpha) \; |||G_{zw}(j\omega, \bar{p})|||_\infty, & \text{else if} \quad C(\bar{p}) < 0. \end{cases} \qquad (13)$$

III. DESIGN OF STRUCTURALLY CONSTRAINED CONTROLLERS

The direct optimisation based approach with structural constrains to selected elements within the controller matrices (A_c, B_c, C_c and D_c) is presented in this section. These constrained or fixed elements are not considered as variables in the optimisation problem.

Let us consider a matrix C_M which contains all the controller gain matrices. This matrix is later vectorised for constructing the vector $\bar{p}$ containing the optimisation variables,

$$C_M = \begin{bmatrix} A_c & \vdots & B_c \\ \overline{C_c} & \vdots & \overline{D_c} \end{bmatrix}.$$

Some of the elements within C_M are fixed, and are not to be considered in the vector $\bar{p}$. This can be portrayed with the help of an example problem of designing overlapping controllers.

Example 4. Let us consider an example MIMO system

for which a second order controller (with an overlapping configuration of C_M) has to be designed as follows,

$$\begin{bmatrix} \dot{x}_{c1} \\ \dot{x}_{c2} \\ u_1 \\ u_2 \end{bmatrix} = \overbrace{\begin{bmatrix} a_{c_{11}} & 0 & \vdots & b_{c_{11}} & \boxed{b_{c_{12}}} \\ 0 & a_{c_{22}} & \vdots & 0 & b_{c_{22}} \\ \hline c_{c_{11}} & 0 & \vdots & d_{c_{11}} & \boxed{d_{c_{12}}} \\ 0 & c_{c_{22}} & \vdots & 0 & d_{c_{22}} \end{bmatrix}}^{C_M} \begin{bmatrix} x_{c1} \\ x_{c2} \\ y_1 \\ y_2 \end{bmatrix}. \qquad (14)$$

This MIMO system has two inputs and two outputs. If b_{c12} and d_{c12} were to be zero elements, we would have decentralised sub-controllers. That is, input, output, and sub-controller state interactions are decoupled. Since b_{c12} and
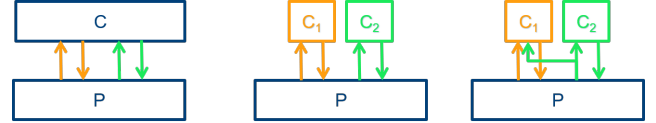

Fig. 1. Overview of centralised, decentralised, and overlapping configurations. P is the MIMO plant with two inputs and two outputs whereas C, C_1, and C_2 are the controllers.

d_{c12} are non-zero elements or not fixed, we have to design overlapping sub-controllers. That is, only input and sub-controller state interactions are decoupled, while, one of the measured output is shared between the sub-controllers. In this example, to optimise the overlapping (or decentralised) sub-controllers without losing its structure, we must keep the 0 elements fixed. The difference between centralised, decentralised and overlapping configurations can be visualised with the help of Fig. 1.

In general, imposing zero values to specific controller parameters could lead to segments (sub-controllers) within one controller having restricted access to certain measured outputs and/or restricted control of certain inputs.

A. Decentralised and overlapping controllers

As mentioned earlier, it is possible to design decentralised and overlapping controllers using the principle of structural constrains. The structural constrains can be enforced on C_M in Example 4 with the help of a matrix F_M.

$$f_{Mij} \begin{cases} 1, & \text{if } c_{Mij} \text{ is an optimisation variable} \\ 0, & \text{else if } c_{Mij} \text{ is a fixed element} \end{cases} \qquad (15)$$

In (15), c_{Mij} and f_{Mij} denote the elements of the ith row and the jth column in the matrices C_M and F_M respectively. By definition, the sizes of the matrices C_M and F_M are identical.

$$\bar{p} = \underset{F_M}{\text{vec}} \, C_M = \underset{F_M}{\text{vec}} \begin{bmatrix} A_c & \vdots & B_c \\ \overline{C_c} & \vdots & \overline{D_c} \end{bmatrix} \qquad (16)$$

Where, $\text{vec}_{F_M} C_M$ is a vector containing the elements of C_M for which the corresponding element in F_M is one, see (15). The elements in $\text{vec}_{F_M} C_M$ and $\text{vec} \, C_M$ are in the same order. We obtain the new controller parameter vector $\bar{p}$ using (16).

For this purpose we can define two interaction matrices M_{C_u} and M_{C_y}, which denote the interaction between input, output

and sub-controllers. We also define a vector $\bar{n}_{C_a}$ to contain information on the order of all the sub-controllers. M_{C_u}, M_{C_y} and $\bar{n}_{C_a}$ are given as input to the algorithm for the design of decentralised or overlapping type of structurally constrained controller. Letting $m_{C_u ij}$ and $m_{C_y ij}$ denote the elements of the ith row and the jth column in matrices M_{C_u} and M_{C_y} respectively, we have

$$m_{C_u ij} \begin{cases} 1, & \text{if } i^{\text{th}} \text{ controller handles the } j^{\text{th}} \text{ input} \\ 0, & \text{otherwise} \end{cases}$$

$$m_{C_y ij} \begin{cases} 1, & \text{if } i^{\text{th}} \text{ controller considers the } j^{\text{th}} \text{ output} \\ 0, & \text{otherwise.} \end{cases}$$

Referring back to Example 4, the input given to the algorithm for designing (14) are given as,

$$M_{C_u} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad M_{C_y} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad \bar{n}_{C_a} = \begin{bmatrix} 1 & 1 \end{bmatrix}^{\text{T}}. \tag{17}$$

Therefore, we consider two first order sub-controllers. We need to fix some elements in the matrix C_M to zero in order to have the same form as the matrix within (14). Subsequently, with the information available in (17), it is also possible to obtain the matrix F_M,

$$F_M = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}. \tag{18}$$

Using (18), we can construct the new C_M as in (14), this is the structurally (or sparsity) constrained form of the controller matrix. The corresponding vector $\bar{p}$ for Example 4 can be given as,

$$\bar{p} = [a_{c_{11}} \ c_{c_{11}} \ a_{c_{22}} \ c_{c_{22}} \ b_{c_{11}} \ d_{c_{11}} \ b_{c_{12}} \ b_{c_{22}} \ d_{c_{12}} \ d_{c_{22}}]^{\text{T}}. \tag{19}$$

We can represent the matrix F_M in general form with the help of the matrix of ones (in what follows, $J_{n \times n}$ denotes the matrix of size n by n with every entry equal to one). If l is the total number of sub-controllers, then $k \in \{1, ..., l\}$ and n_{c_k} is the order of the kth sub-controller. If the total number of inputs is w, then $h \in \{1, ..., w\}$. Similarly, when the total number of outputs is z, then $j \in \{1, ..., z\}$. For Example 4 with input as in (17), there are two sub-controllers, two inputs and two outputs, then $l = 2$, $w = 2$ and $z = 2$ respectively. The general representations for matrices $J_{n \times n}$ and F_M are given below.

$$F_M = \begin{bmatrix} J_{n_{c_1} \times n_{c_1}} & \cdots & 0 & \vdots & \\ \vdots & \ddots & \vdots & \vdots & \left[m_{C_y kj} \cdot J_{n_{c_k} \times 1}\right]_{k,j} \\ 0 & \cdots & J_{n_{c_l} \times n_{c_l}} & \vdots & \\ \hline \left[m_{C_u hk} \cdot J_{1 \times n_{c_k}}\right]_{h,k} & & & \vdots & M_{C_u} M_{C_y} \end{bmatrix}.$$

Here we use $[\ \cdot\ ]_{i,j}$ to denote the (i, j)-th block of a matrix. For both the cases of overlapping and decentralised

controllers A_c takes a block diagonal form as shown below.

$$A_c = \begin{bmatrix} A_{c1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & A_{cl} \end{bmatrix}$$

The matrices B_c, C_c and D_c will be sparsity constrained but they need not be block diagonal in structure. Also, this could be the case for decentralised configuration. Sparsity constrains are defined based on the interaction matrices and the order of the sub-controllers. Subsequently, the interaction matrices M_{C_u} and M_{C_y} which are not of diagonal form will result in controller gain matrices B_c, C_c, and D_c which are not of block diagonal form. However, this does not restrict the implementation of this tool in anyway.

B. Other controllers

One can use the concept of structural constraints to design many other controllers. A kind of distributed controller can be considered by including the off-diagonal elements of the A_c matrix in the vector $\bar{p}$. PID controllers are commonly used as feedback controllers in the industry. It is also possible to structurally constrain the dynamic controller to represent a PID controller and optimise its gains. Let us consider the PID controller mentioned in [11].

$$K(s) = K_P + K_I \frac{1}{s} + K_D \frac{s}{1 + \tau_d s}, \tag{20}$$

for which a realisation is determined by the controller matrices,

$$\left[\begin{array}{c|c} A_c & B_c \\ \hline C_c & D_c \end{array}\right] = \left[\begin{array}{cc|c} 0 & 0 & K_i \\ 0 & -\frac{1}{\tau_d} I & -\frac{1}{\tau_d^2} K_d \\ \hline I & I & K_p + \frac{1}{\tau_d} K_d \end{array}\right] \tag{21}$$

Here τ_d is the time constant of the filter applied to the derivative action. The physical reliability is safeguarded by ensuring the properness of the PID controller using this low-pass first order filter (see [11]). If we assume τ_d to be a constant, we can convert this into an optimisation problem for the proposed algorithm as given below.

$$F_M = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \rightarrow C_M = \begin{bmatrix} 0 & 0 & b_{c_{11}} \\ 0 & -\frac{1}{\tau_d} I & b_{c_{21}} \\ I & I & d_{c_{11}} \end{bmatrix} \mapsto \bar{p} = \begin{bmatrix} b_{c_{11}} \\ b_{c_{21}} \\ d_{c_{11}} \end{bmatrix}$$

The new values for the gains of the PID controller can be obtained from the optimised dynamic controller using $K_i = b_{c_{11}}$, $K_d = -\tau_d^2 b_{c_{21}}$ and $K_p = d_{c_{11}} - \frac{1}{\tau_d} K_d$.

IV. EXAMPLE MIMO PROBLEMS

TABLE I
INFORMATION ON THE EXAMPLE TDS CONSIDERED

Example	Order of plant	No. of inputs	No. of outputs	No. of time-delays
Neutral TDS	3	2	2	5
Retarded TDS	4	2	2	1

In this section, two MIMO plants with time-delays are used by the proposed algorithm to obtain structurally constrained controllers. Some basic information on the structure

Example	Structurally constrained controller	Objective function	$\mathcal{H}_\infty$ norm	Spectral abscissa
Neutral TDS	—	—	∞	0.6836
	Overlapping ($n_c = 1 + 3$)	RSA ($\alpha = 1$)	468.73	−0.2482
	Overlapping ($n_c = 1 + 3$)	SHN ($\alpha = 0$)	149.19↓	−0.1558↑
	Decentralised ($n_c = 1 + 3$)	RSA ($\alpha = 1$)	2,593.43	−0.0888
	Decentralised ($n_c = 1 + 3$)	SHN ($\alpha = 0$)	1,778.52↓	−0.0753↑
Retarded TDS	—	—	∞	0.7990
	Overlapping ($n_c = 2 + 3$)	RSA ($\alpha = 1$)	2.10	−0.3741
	Overlapping ($n_c = 2 + 3$)	SHN ($\alpha = 0$)	1.68↓	−0.2632↑
	Decentralised ($n_c = 2 + 3$)	RSA ($\alpha = 1$)	31.83	−0.3041
	Decentralised ($n_c = 2 + 3$)	SHN ($\alpha = 0$)	26.07↓	−0.0939↑

of these plants are given in Table I.

The results obtained for the closed-loop systems of the plants and decentralised or overlapping controllers are shown in Table II. Only the final results have been presented in the table due to the space limitation[1]. In both these example problems, when $\alpha = 1$ controllers were optimised for minimum robust spectral abscissa (RSA). However, when $\alpha = 0$ the controllers were optimised for minimum strong $\mathcal{H}_\infty$ norm (SHN). For the examples considered in this paper, we can observe that minimisation of the strong $\mathcal{H}_\infty$ norm occurs at the cost of reduced exponential decay rate (an increase in the value of robust spectral abscissa). Also, we can observe that the overlapping controllers generally perform better than the decentralised controllers which is expected since they result in less structural constraints on the controller parameters.

V. CONCLUSION

In this paper, a methodology to design structurally constrained dynamic (LTI) controllers was presented. It was concluded that decentralised controllers, overlapping controllers and many other types of controllers can be considered as structurally constrained controllers, for which a generic design approach was presented. The proposed frequency domain based approach was used to design stabilising and robust fixed-order decentralised and overlapping controllers for linear time invariant neutral and retarded time-delay systems.

The approach has been implemented as an improvement to the algorithms in [2] and [3], therefore, the objective functions are in general non-convex. This is addressed by using randomly generated initial values for controller parameters, along with initial controllers specified by the user, and choosing the most optimal solution from them.

The algorithm presented here relies on a routine for computing the objective function and its gradient whenever the objective function is differentiable. For the spectral abscissa, the value of the objective function is obtained by computing rightmost eigenvalues of the DDAE. The value for $\mathcal{H}_\infty$ norm is obtained by a generalisation of the Boyd-Balakrishnan-Kabamba / Bruinsma-Steinbuch algorithm relying on computing imaginary axis solutions of an associated Hamiltonian eigenvalue problem. Evaluating the value of the objective function at every iteration constitutes the dominant computational cost. On the contrary, the derivatives with respect to the controller parameters are computed at a negligble cost from left and right eigenvectors. Due to this and the fact that controllers of lower order are desirable for application, introducing structural constraints will not have a considerable impact on the overall computational complexity of the control design problem.

REFERENCES

[1] D. Siljak, *Decentralized Control of Complex Systems*, ser. Dover Books on Electrical Engineering. Dover Publications, 2013. [Online]. Available: https://books.google.be/books?id=FdXCAgAAQBAJ

[2] S. Gumussoy and W. Michiels, "Fixed-order h-infinity control for interconnected systems using delay differential algebraic equations," *SIAM Journal on Control and Optimization*, vol. 49, no. 5, pp. 2212–2238, 2011. [Online]. Available: http://dx.doi.org/10.1137/100816444

[3] W. Michiels, "Spectrum-based stability analysis and stabilisation of systems described by delay differential algebraic equations," *IET Control Theory Applications*, vol. 5, no. 16, pp. 1829–1842, November 2011.

[4] J. Burke, D. Henrion, A. Lewis, and M. Overton, "HIFOO - a matlab package for fixed-order controller design and H-infinity optimization," *IFAC Proceedings Volumes*, vol. 39, no. 9, pp. 339 – 344, 2006, 5th IFAC Symposium on Robust Control Design. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1474667015335229

[5] M. L. Overton, "HANSO: a hybrid algorithm for nonsmooth optimization. available from http://cs.nyu.edu/overton/software/hanso/," *Computer Science, New York University*, 2009. [Online]. Available: http://cs.nyu.edu/overton/software/hanso/

[6] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis, "State-space solutions to standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control problems," *IEEE Transactions on Automatic Control*, vol. 34, no. 8, pp. 831–847, Aug 1989.

[7] P. Gahinet and P. Apkarian, "A linear matrix inequality approach to h control," *International Journal of Robust and Nonlinear Control*, vol. 4, no. 4, pp. 421–448, 1994. [Online]. Available: http://dx.doi.org/10.1002/rnc.4590040403

[8] S. Sojoudi, J. Lavaei, and A. Aghdam, *Structurally Constrained Controllers: Analysis and Synthesis*, ser. SpringerLink : Bücher. Springer US, 2011. [Online]. Available: https://books.google.be/books?id=daxlGWYeA6gC

[9] A. Alavian and M. C. Rotkowitz, "Q-parametrization and an sdp for hinf-optimal decentralized control," *IFAC Proceedings Volumes*, vol. 46, no. 27, pp. 301 – 308, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1474667015402423

[10] S. M. Ozer and A. Iftar, "Simultaneous decentralized controller design for time-delay systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6495 – 6500, 2017, 20th IFAC World Congress. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2405896317309850

[11] R. Toscano, *Structured Controllers for Uncertain Systems: A Stochastic Optimization Approach*, ser. Advances in Industrial Control. Springer London, 2013. [Online]. Available: https://books.google.be/books?id=qf3hSresSfcC

---

[1]Please refer `http://twr.cs.kuleuven.be/research/ software/delay-control/structurallyconstrainedTDS. zip` to obtain the tool and more information on example problems and their solutions.