

# MEASUREMENT-SCHEDULED CONTROL FOR THE RTAC PROBLEM: AN LMI APPROACH

STÉPHANE DUSSY\* AND LAURENT EL GHAOUI

*Ecole Nationale Supérieure de Techniques Avancées, 32, Blvd. Victor, 75739 Paris, France*

## SUMMARY

The RTAC nonlinear control design benchmark problem is addressed using a multi-objective control methodology based on linear matrix inequalities and robust control. The approach hinges on the search for a common quadratic Lyapunov function ensuring various specifications (stability,  $\mathcal{L}_2$ -gain, command input and output peak bounds) for the closed-loop system. The resulting output-feedback controller is measurement-scheduled; precisely, its state-space matrices depend on the measurement vector, in a nonlinear fashion. We evaluate the performance of our design with various simulations and predicted trade-off curves. © 1998 John Wiley & Sons, Ltd.

Key words: linear matrix inequality; quadratic Lyapunov function; robust nonlinear control; linear-fractional representation; gain-scheduling

## 1. INTRODUCTION

### 1.1. Purpose

In this paper, we concentrate on a *multicriteria nonlinear control problem*, where a controller is sought, such that a number of possibly conflicting constraints are met by the closed-loop system.

The paper is intended to serve two purposes. First, we seek to provide a tutorial on a systematic methodology for multicriteria nonlinear control problems, recently developed in References 1 and 2. Second, we illustrate this methodology on the RTAC benchmark problem introduced by Bupp *et al.*<sup>3</sup>

Several approaches of the RTAC problem have been proposed in the past few years. Various stabilization strategies are compared in Reference 4. Partial feedback linearization and integrator backstepping methods are studied in References 5 and 6; adequate Lyapunov functions lead to a backstepping controller in Reference 7 and to a passive nonlinear controller in Reference 3. A state-feedback nonlinear controller is obtained in Reference 8, via the solution of a Hamilton–Jacobi–Isaacs equation.

Nonlinear control design is by itself a whole field of study, and a great variety of methods have been proposed: the second method of Lyapunov,<sup>9,10</sup> exact linearization,<sup>11</sup> extension of  $\mathbf{H}_\infty$  methodology to nonlinear system,<sup>12–14</sup> etc. Most of these techniques do not address multicriteria

\*Correspondence to: Stéphane Dussy, Ecole Nationale Supérieure de Techniques Avancées, 32, Blvd. Victor, 75739 Paris, France. E-mail: dussy@ensta.fr.

problems directly. Furthermore, they may lead to equations that are hard to solve numerically, in particular when output-feedback robust synthesis is involved. Finally, most of these approaches do not take into account uncertainty, nor multiple specifications, explicitly.

The methodology proposed here is based on the search of a ‘common’ (quadratic) Lyapunov function for the closed-loop system, that guarantees the various design constraints simultaneously.

Our choice of a common, quadratic Lyapunov function implies some degree of conservatism of the method. This disadvantage is traded-off against the numerical tractability of the resulting design algorithms. In fact, the method is a way to reformulate the control problem as one of choosing a few design parameters with direct physical interpretation (such as degree of stability,  $L_2$  gain, etc.). (A more detailed discussion on common Lyapunov functions is in Section 2.2.)

It is, of course, possible to extend our approach to more elaborate (less-conservative) Lyapunov functions, using, for example, parameter-dependent Lyapunov functions,<sup>15,16</sup> integral quadratic constraints,<sup>17</sup> frequency-dependent multipliers,<sup>18</sup> Lur’e-Postnikov-type functions<sup>19,20</sup> or parameterized Lyapunov bounds.<sup>21</sup> However, most of these results only apply for robust analysis—they lead to very hard non-convex problems for robust synthesis. For example, attacking the robust synthesis problem with parameter-dependent Lyapunov functions leads to a set of bilinear matrix inequalities (BMIs).

In contrast, using quadratic Lyapunov functions allows a numerically easier solution to synthesis problems. (Recall that our main objective is to have a systematic and numerically tractable method.)

Based on a ‘linear-fractional representation’<sup>22–24</sup> of the open-loop nonlinear system, we use robust control approaches and linear matrix inequality (LMI) optimization.<sup>25</sup> The method is an extension of the gain-scheduling approach of Packard,<sup>26,27</sup> to systems with non-necessarily bounded nonlinearities. To obtain the controller, we must find a solution to a set of LMI conditions, assorted with (at most) one quadratic, non-convex constraint. Our formulation enables us to use a cone-complementarity linearization algorithm that has proved its efficiency in the static output-feedback problem,<sup>28</sup> and in several other nonlinear control problems, where alternative (BMI-based) approaches fail.

## 1.2. Paper outline

In Section 2, we provide an overview on our methodology. We describe in Section 3 the mathematical specifications for our control design problem and we state it in terms of linear-fractional representations (LFRs). LMI-based synthesis conditions are derived in Section 4. The linear-fractional representation of the RTAC benchmark system, as described in Reference 3, and related numerical results, are given in Section 5.

## 1.3. Notations

For a real matrix  $P$ ,  $P > 0$  (resp.  $P \geq 0$ ) means  $P$  is symmetric and positive-definite (resp. positive-semi-definite);  $P^{1/2}$  denotes its symmetric square root. The notation  $\mathbf{diag}(A, B)$  with  $A \in \mathbb{R}^{p \times q}$  and  $B \in \mathbb{R}^{m \times n}$  denotes the matrix

$$\mathbf{diag}(A, B) = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix}$$

For a given integer vector  $r = [r_1, \dots, r_n]$ ,  $r_i \in \mathbb{N}$ ,  $i = 1, \dots, n$ , we define the sets

$$\mathcal{D}(r) = \{\Delta = \mathbf{diag}(\delta_1 I_{r_1}, \dots, \delta_n I_{r_n}) \mid \delta_i \in \mathbb{R}, i = 1, \dots, n\}$$

$$\mathcal{B}(r) = \{B = \mathbf{diag}(B_1, \dots, B_n) \mid B_i \in \mathbb{R}^{r_i \times r_i}, i = 1, \dots, n\}$$

$$\mathcal{S}(r) = \{S \in \mathcal{B}(r) \mid S_i > 0, i = 1, \dots, n\}$$

$$\mathcal{G}(r) = \{G \in \mathcal{B}(r) \mid G_i + G_i^T = 0, i = 1, \dots, n\}$$

Finally,  $\mathbf{Co}\{v_1, \dots, v_L\}$  denotes the polytope with vertices  $v_1, \dots, v_L$ .

## 2. AN OVERVIEW OF THE METHODOLOGY

### 2.1. Principles

The basic problem we consider is to find a nonlinear controller for a given non-linear system, such that the closed-loop system meets a number of specifications. (We will be more precise about the required assumptions later.) The method is based on LFRs, quadratic Lyapunov functions and multicriteria robust control design.

First, we derive a LFR for the open-loop system. In this representation, the system is viewed as a linear, time-invariant system connected to a feedback matrix  $\Delta$  that depends on (some of) the state variables (see Figure 1). The LFR is then used in conjunction with robust control approaches, to analyse and synthesize control laws for the system. Note that robust control methods cannot in general be used directly, since the feedback matrix is usually not bounded *a priori*.

The analysis is based on the search for a common quadratic Lyapunov functions, that is required to prove the desired specifications (for example, to guarantee that a certain region around the equilibrium point 0 is stable). The synthesis problem is to find both a measurement-scheduled control law, and a quadratic Lyapunov function proving that the closed-loop system meets the specifications. The control law is searched in the form of an LFR, whose coefficients are to be chosen. The optimization problem for controller synthesis thus involves two kinds of variables: some are related to the quadratic Lyapunov function, and the others to the coefficients of the LFR which determine the control law.

When the open-loop system is linear in the variables that are not measured, the feedback matrix  $\Delta(x)$  in the LFR depends only on the measured output  $y$ . In this case, the method leads to a convex optimization problem with LMI constraints.

When the open-loop system contains non-measured nonlinearities, the method yields optimization problems with LMI constraints, and additional rank constraints. This kind of problem is

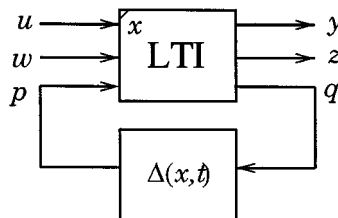


Figure 1. Linear-fractional representation of the open-loop system

not convex; however an efficient heuristic, based on solving a sequence of LMI problems, can be used in this case.

There are several key features in the proposed approach.

- *Efficient numerical solution.* LMI optimization problems can be solved very efficiently using recent interior-point methods (the global optimum is found in modest computing time). This brings a numerical solution to problems when no analytical or closed-form solution is known.
- *A systematic method.* The approach is very systematic and can be applied to a very wide variety of nonlinear control design problems. This includes (and is not reduced to) systems whose state-space representations are rational functions of the state vector.
- *Extension to uncertain nonlinear systems.* It is possible to extend the method to cases when some parameters in the state-space representation of the system are uncertain.
- *Multicriteria problems.* The approach is particularly well suited to problems where several (possibly conflicting) specifications are to be satisfied by the closed-loop system. This is made possible by use of a common Lyapunov function *proving* that every specification holds.

In view of the above features, the approach opens the way to CAD tools for the analysis and control of nonlinear systems, see, e.g. the public domain toolbox MRCT<sup>29,30</sup> built on top of the software lmitool.<sup>31</sup>

## 2.2. Common Lyapunov functions in multicriteria design

Using the same Lyapunov function to enforce different specifications may lead to conservative results. This disadvantage is traded off in several ways.

First, other methods, such as  $H_\infty$ -control, do not allow to impose *directly* several constraints. One must form a single criterion, in the form of an  $H_\infty$ -norm bound, that is deemed to reflect all specifications. Forming this one criterion may be difficult, since a (sometimes large) number of parameters have to be chosen. Also, these design parameters do not always have obvious relationship with the desired constraints.

In an LMI-based design, we form a set of LMI constraints, each one reflecting one specification. In each constraint appears a parameter that determines how coercive the specification is. Increasing the parameter amounts to relax the constraint. For example, if we impose an upper bound  $u_{\max}$  on the command input norm, then  $u_{\max}$  is a measure of how stringent the norm bound is. The various parameters (command input  $u_{\max}$ ,  $\mathcal{L}_2$ -gain bound  $\gamma$ , etc.) can be interpreted as *design parameters*.

In order to meet the specifications, the approach thus reduces the synthesis problem to choosing a *few* design parameters. Note that all the design parameters have a direct physical interpretation, and that there is only a small number of them.

The possible conservatism of the LMI approach can be reduced if we choose the design parameters iteratively, as follows. At the first step of the design process, we may see the various parameters to higher values than those imposed by the actual specifications. In a second step, we may perform an (LMI-based) *analysis*, or a simulation, on the closed-loop system, in order to check if the latter obeys the specifications. (Since LMI-based design tools are more conservative than LMI-based analysis tools, it seems that this will be often the case.) If not, we can identify (using analysis or simulation results) which specifications are violated, and modify the corresponding design parameters accordingly.

## 3. PROBLEM STATEMENT

## 3.1. Open-loop system

We consider a nonlinear, time-invariant, continuous-time system with a zero equilibrium point:

$$\begin{aligned}\dot{x} &= \mathbf{A}(\delta(x, t))x + \mathbf{B}_u(\delta(x, t))u + \mathbf{B}_w(\delta(x, t))w \\ y &= \mathbf{C}_y(\delta(x, t))x + \mathbf{D}_{yu}(\delta(x, t))u \\ z &= \mathbf{C}_z(\delta(x, t))x + \mathbf{D}_{zw}(\delta(x, t))w\end{aligned}\quad (1)$$

where  $x \in \mathbb{R}^n$  is the state vector,  $u \in \mathbb{R}^{n_u}$  is the command input,  $w \in \mathbb{R}^{n_w}$  is the exogenous input,  $z \in \mathbb{R}^{n_z}$  is the output of interest and  $y \in \mathbb{R}^{n_y}$  is the measured output. The vector-valued function  $\delta \in \mathbb{R}^L$  is a nonlinear function of state  $x$  and time  $t$ .

We make the following assumptions.

- A.1.  $\mathbf{A}$ ,  $\mathbf{B}_u$ ,  $\mathbf{C}_y$ ,  $\mathbf{C}_z$ ,  $\mathbf{D}_{yu}$  and  $\mathbf{D}_{zw}$  are rational functions of their argument, that are well defined at  $x = 0$ .
- A.2. The function  $(x, t) \rightarrow \delta(x, t) \in \mathbb{R}^L$  is Lipschitz, i.e. there exist constants  $M_1$ ,  $M_2$  such that, for every  $x$ ,  $t$ ,  $\|\delta(x, t)\| \leq M_1\|x\| + M_2$ .

(Note that assumption A.2 can be relaxed).

Using the methodology described in References 1 and 2, which we briefly recall in Appendix A, we can derive from (1) an LFR for the system

$$\begin{aligned}\dot{x} &= Ax + B_u u + B_p p + B_w w \\ z &= C_z x + D_{zp} p + D_{zu} u + D_{zw} w \\ y &= C_y x + D_{yp} p + D_{yu} u + D_{yw} w \\ q &= C_q x + D_{qp} p + D_{qu} u + D_{qw} w \\ p &= \Delta(x(t), t)q\end{aligned}\quad (2)$$

where

$$\Delta(x, t) = \mathbf{diag}(\delta_1(x, t)I_{r_1}, \dots, \delta_L(x, t)I_{r_L})$$

and  $r = [r_1 \dots r_L]$  is an integer vector (the element  $r_i$  is related to the highest degree of  $\delta_i$  in the state-space functions of the system).

With the above representation, the system is viewed as a feedback connection between an LTI system and a (nonlinear) matrix  $\Delta(x, t)$  (see Figure 1).

For further reference, we now distinguish in the feedback matrix  $\Delta$  the elements that are measured in real time (we will label those **meas**, for ‘measured’), from those that are not measured (labelled **unk**, for ‘unknown’). Without loss of generality, we may assume that the first  $M$  elements of  $\delta$  are measured:

$$\Delta(x, t) = \mathbf{diag}(\Delta_{\text{meas}}(y), \Delta_{\text{unk}}(x, t))$$

where  $\Delta_{\text{meas}}(y) = \mathbf{diag}(\delta_1(y)I_{r_1}, \dots, \delta_M(y)I_{r_M})$  depends only on the measurement vector  $y$ .

For later reference, let  $v_{\text{unk}}$  be the size of  $\Delta_{\text{unk}}$ , and  $v_{\text{meas}}$  be the size of  $\Delta_{\text{meas}}$ . The total size of the  $\Delta$  matrix is  $v = v_{\text{unk}} + v_{\text{meas}}$ .

3.2. *Controller structure*

The controller structure is chosen to be ‘measurement-scheduled’, i.e.

$$\begin{aligned} \dot{\bar{x}} &= \bar{\mathbf{A}}(\Delta_{\text{meas}}(y))\bar{x} + \bar{\mathbf{B}}_y(\Delta_{\text{meas}}(y))y, \quad \bar{x}(0) = 0 \\ u &= \bar{\mathbf{C}}_u\bar{x} \end{aligned} \tag{3}$$

where  $\bar{x}$  is the controller state,  $\bar{\mathbf{C}}_u$  is a constant matrix and  $\bar{\mathbf{A}}, \bar{\mathbf{B}}_y$  are rational functions of their arguments  $\delta_i(y), i = 1, \dots, M$ . (The term measurement-scheduled comes from the fact that, when  $y$  is fixed, the above controller is linear time-invariant.) We shall also consider the case when the full state is measured. In this case, our method will find a static, state-feedback control of the form

$$u = \bar{\mathbf{C}}_u x$$

where  $\bar{\mathbf{C}}_u$  is a constant matrix.

Assuming that the matrix-valued functions  $\bar{\mathbf{A}}(\cdot), \bar{\mathbf{B}}_y(\cdot)$ , etc. are rational functions of their argument, we can always assume our measurement-scheduled controller to be in the LFR format:

$$\begin{aligned} \dot{\bar{x}} &= \bar{\mathbf{A}}\bar{x} + \bar{\mathbf{B}}_y y + \bar{\mathbf{B}}_p \bar{p}, \quad \bar{x}(0) = 0 \\ \bar{q} &= \bar{\mathbf{C}}_q \bar{x} + \bar{\mathbf{D}}_{qy} y + \bar{\mathbf{D}}_{qp} \bar{p} \\ u &= \bar{\mathbf{C}}_u \bar{x} \\ \bar{p} &= \Delta_{\text{meas}}(y)\bar{q} \end{aligned} \tag{4}$$

where  $\bar{p}$  and  $\bar{q}$  are fictitious inputs and outputs.

We recall the crucial assumption here: the controller is scheduled with respect to the measured element in the nonlinear block  $\Delta$ , namely  $\Delta_{\text{meas}}(y)$ .

3.3. *LFR of the closed-loop system*

With the chosen controller structure, the closed-loop system assumes the form

$$\dot{\tilde{x}} = \tilde{\mathbf{A}}(\tilde{\Delta}(x, t))\tilde{x} + \tilde{\mathbf{B}}_w(\tilde{\Delta}(x, t))w \tag{5}$$

where  $\tilde{x} = [x^T \ \bar{x}^T]^T$ . Here,  $\tilde{\mathbf{A}}, \tilde{\mathbf{B}}_w$  are given rational functions of the nonlinearity element  $\tilde{\Delta}$ , where

$$\tilde{\Delta}(x, t) = \mathbf{diag}(\Delta_{\text{meas}}(y), \Delta_{\text{unk}}(x, t), \Delta_{\text{meas}}(y)) \tag{6}$$

Introducing

$$\tilde{p} = \begin{bmatrix} p \\ \bar{p} \end{bmatrix}, \quad \tilde{q} = \begin{bmatrix} q \\ \bar{q} \end{bmatrix}$$

we obtain the following LFR for the closed-loop system:

$$\begin{aligned} \dot{\tilde{x}} &= (\tilde{\mathbf{A}} + \tilde{\mathbf{B}}\tilde{\mathbf{K}}\tilde{\mathbf{C}}_y + \tilde{\mathbf{B}}_u\tilde{\mathbf{K}}_u)\tilde{x} + (\tilde{\mathbf{B}}_p + \tilde{\mathbf{B}}\tilde{\mathbf{K}}\tilde{\mathbf{D}}_{yp})\tilde{p} + (\tilde{\mathbf{B}}_w + \tilde{\mathbf{B}}\tilde{\mathbf{K}}\tilde{\mathbf{D}}_{yw})w \\ \tilde{q} &= (\tilde{\mathbf{C}}_q + \tilde{\mathbf{D}}\tilde{\mathbf{K}}\tilde{\mathbf{C}}_y + \tilde{\mathbf{D}}_{qu}\tilde{\mathbf{K}}_u)\tilde{x} + (\tilde{\mathbf{D}}_{qp} + \tilde{\mathbf{D}}\tilde{\mathbf{K}}\tilde{\mathbf{D}}_{yp})\tilde{p} + (\tilde{\mathbf{D}}_{qw} + \tilde{\mathbf{D}}\tilde{\mathbf{K}}\tilde{\mathbf{D}}_{yw})w \\ z &= (\tilde{\mathbf{C}}_z + \tilde{\mathbf{D}}_{zu}\tilde{\mathbf{K}}_u)\tilde{x} + \tilde{\mathbf{D}}_{zp}\tilde{p} \\ \tilde{p} &= \tilde{\Delta}\tilde{q}, \quad \tilde{\Delta}(x, t) = \mathbf{diag}(\Delta_{\text{meas}}(y), \Delta_{\text{unk}}(x, t), \Delta_{\text{meas}}(y)) \end{aligned} \tag{7}$$

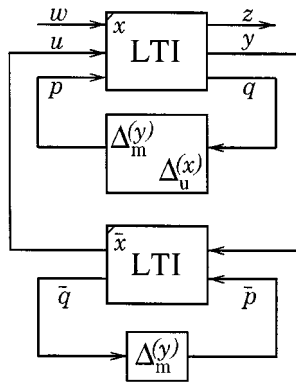


Figure 2. LFR of the closed-loop system

where the matrices  $\tilde{A}$ ,  $\tilde{B}_p$ , etc., defined in Appendix B, depend affinely on the design matrix variables  $\bar{C}_u$  and  $K$ . The closed-loop system is shown in Figure 2.

In the above, the matrices  $\bar{C}_u$  and

$$K = \begin{bmatrix} \bar{A} & \bar{B}_p & \bar{B}_y \\ \bar{C}_q & \bar{D}_{qp} & \bar{D}_{qy} \end{bmatrix}$$

are the design variables.

### 3.4. Control design specifications

Typical specifications in a control design problem are

- The closed-loop system is stable over a given region containing 0.
- The closed-loop system exhibits good settling behaviour for a class of initial conditions.
- The closed-loop system exhibits good disturbance rejection compared to the uncontrolled system for a class of disturbance signals.
- The control effort should be reasonable.
- Some output of interest  $z$  should not exceed a given level.

We now translate the above specifications in a mathematical form, which is defined in terms of several ‘design parameters’.

First, to a given vector  $x_0$  with non-negative elements, we associate a region of admissible initial conditions, a polytope of the form

$$\mathcal{P}(x_0) = \text{Co} \left\{ \begin{bmatrix} \pm x_{0,1} \\ \vdots \\ \pm x_{0,n} \end{bmatrix} \right\} \tag{8}$$

In the sequel, we denote by  $v_j, j = 1, \dots, 2^n$ , the vertices of  $\mathcal{P}(x_0)$ . (It is possible to work with ellipsoidal sets of initial conditions, such as  $\mathcal{P}(x_0) = \{x_0 \mid \|x_0\| \leq 1\}$ .) We also consider the set of

admissible disturbances, chosen to be of the form

$$\mathcal{W}(w_{\max}) = \left\{ w \in \mathcal{L}_2([0, \infty]) \left| \int_0^\infty w(t)^T w(t) dt \leq w_{\max}^2 \right. \right\}$$

where  $w_{\max}$  is a given scalar.

To the sets  $\mathcal{P}(x_0)$  and  $\mathcal{W}(w_{\max})$  we associate the set of ‘test trajectories’ for the closed-loop system, i.e.

$$\mathcal{X}(x_0, w_{\max}) = \left\{ \tilde{x} = \begin{bmatrix} x \\ \bar{x} \end{bmatrix} \left| \begin{array}{l} \tilde{x} \text{ satisfies (5),} \\ \bar{x}(0) = 0, x(0) \in \mathcal{P}(x_0), w \in \mathcal{W}(w_{\max}) \end{array} \right. \right\}$$

We introduce a vector  $x_{\max}$  which will be used to bound the state variables ( $x_{\max}$  is a design parameter). We assume that the polytope  $\mathcal{P}(x_0)$  (defined as in (8)) is included in the polytope  $\mathcal{P}(x_{\max})$ ; in other words,  $x_{i,0} \leq x_{i,\max}$ ,  $i = 1, \dots, n$ . Finally, let  $\alpha$ ,  $u_{\max}$  and  $\gamma$  be positive scalars.

Our design constraints are as follows.

- S.1. The system is well posed, to be precise, the closed-loop system’s equations (5) are well defined for every  $x \in \mathcal{P}(x_{\max})$ .
- S.2. Every trajectory in  $\mathcal{X}(x_0, 0)$  decays to zero at rate  $\alpha$ , i.e.

$$\lim_{t \rightarrow \infty} e^{\alpha t} \tilde{x}(t) = 0$$

- S.3. For every trajectory in  $\mathcal{X}(x_0, w_{\max})$ , the command input  $u$  satisfies

$$\|u(t)\|_2 \leq u_{\max} \quad \text{for every } t \geq 0$$

- S.4. For every trajectory in  $\mathcal{X}(x_0, w_{\max})$ , the state is bounded componentwise, i.e.

$$|x_i(t)| \leq x_{i,\max} \quad \text{for every } i, 1 \leq i \leq n \text{ and } t \geq 0$$

- S.5. For every trajectory in  $\mathcal{X}(0, w_{\max})$ , and for every  $T \geq 0$ , we have

$$\int_0^T z(t)^T z(t) dt \leq \gamma^2 \int_0^T w(t)^T w(t) dt$$

where  $z$  is a given output.

Our final design depends obviously on the choice of the parameters  $x_0$ ,  $w_{\max}$ ,  $\alpha$ ,  $u_{\max}$ ,  $x_{\max}$  and  $\gamma$ . (Some of these parameters can be taken to be zero or infinite if the corresponding constraint is void.) We stress that the design parameters are not necessarily those imposed by the specifications.

The design parameters have obvious physical interpretations, and they may be adjusted on a heuristic basis. To this end, it is helpful to obtain trade-off curves between pairs of design parameters. For instance, in Section 5, we show curves of achievable performance (measured by  $\gamma$ ) versus control effort (measured by  $u_{\max}$ ), for various values of decay rate  $\alpha$ .

The control design specification S.4 requires the state variables to be bounded componentwise. In view of assumption A.2, this will imply bounds on the matrix  $\Delta$ . Note that these bounds are not necessarily *a priori* bounds: they depend on the design parameter  $x_{\max}$ .



#### 4. SYNTHESIS CONDITIONS

##### 4.1. Analysis via quadratic Lyapunov functions

We seek quadratic Lyapunov functions ensuring specifications S.1–S.5 for the closed-loop system. We thus consider a matrix  $\tilde{P} \in \mathbb{R}^{2n}$ , with  $\tilde{P} > 0$ , and associated Lyapunov function

$$V(\tilde{x}) = \tilde{x}^T \tilde{P} \tilde{x}$$

Assume that, for a given non-zero trajectory in the set  $\mathcal{X}(x_0, w_{\max})$ ,  $V$  satisfies

$$\dot{V} + 2\alpha V - \gamma^2 w^T w + z^T z < 0 \quad (9)$$

The above inequality is at the basis of our control methodology. It obviously guarantees specification S.2. In addition, the above inequality also holds when  $\alpha = 0$ . Integrating both sides of the resulting inequality yields

$$V(\tilde{x}(T)) + \int_0^T z(t)^T z(t) dt \leq V(\tilde{x}(0)) + \gamma^2 \int_0^T w(t)^T w(t) dt \quad (10)$$

The above inequality guarantees S.5. Furthermore, it enables us to bound the state (or any linear combination of it). For example, if  $x(0)$  is chosen such that

$$V(\tilde{x}(0)) \leq \lambda \quad (11)$$

for some positive number  $\lambda$ , then the corresponding trajectory satisfies

$$V(\tilde{x}(T)) \leq \lambda + \gamma^2 w_{\max}^2. \quad (12)$$

It suffices to ensure that inequality (11) holds at every vertex of the polytope of admissible initial conditions  $\mathcal{P}(x_0)$ , to obtain a bound on the resulting state vector that is valid for every trajectory in  $\mathcal{X}$  (specification S.4). We can use this inequality to bound the command input  $u$ , seen as a linear combination of the closed-loop system's state (specification S.3). For more details on this method, we refer to References 1, 2 and 25.

##### 4.2. Related control strategies

Consider the following special cases of (9), namely,

$$\dot{V} + 2\alpha V < 0 \quad (13)$$

and

$$\dot{V} - \gamma^2 w^T w + z^T z < 0 \quad (14)$$

We can outline two alternative control strategies related to the basic inequality (9).

- The *switching strategy* consists in designing two control laws: one is designed to guarantee a return to the vicinity of the equilibrium point in reasonable time, control effort and output bounds despite disturbances, and is based on constraint (13). The other control law is on (14), and is designed to ensure a good disturbance rejection. The overall control law consists in applying the first control until the state reaches a neighbourhood of  $x = 0$ , then switching to the other control law.

- The *mixed strategy* consists in designing one control law obeying to two requirements of stability and disturbance rejection *independently*. This design is based on enforcing (13) and (14) simultaneously, with the same  $V$ .

The basic, switching and mixed all have advantages and drawbacks. The basic strategy might be more conservative than the two others, but it offers guarantees. In contrast, the (less-conservative) mixed strategy offers no guarantee, if the disturbance acts during a stabilization phase. The switching strategy is even less conservative, but raises the additional problem of estimating an appropriate switching time (this problem is easily dealt with if the full state is measured).

4.3. *Matrix inequality conditions for analysis*

In this section, we seek sufficient conditions guaranteeing that the closed-loop system satisfies the design specifications. To describe the basic principle, consider a nonlinear system

$$\dot{x} = \mathbf{A}(\Delta(x))x, \quad \Delta(x) = \mathbf{diag}(x_1, \dots, x_n) \tag{15}$$

Assume we find a quadratic Lyapunov function  $V(x) = x^T P x$  that guarantees robust stability of the associated uncertain system

$$\dot{x} = \mathbf{A}(\Delta(t))x, \quad \Delta \text{ diagonal}, \quad \|\Delta(t)\| \leq 1$$

If, in addition, this Lyapunov function proves that the output  $x_i$  satisfies the bound  $|x_i(t)| \leq x_{i,\max}$  for every  $i$  and that these bounds imply  $\|\Delta(t)\| \leq 1$  for every  $t \geq 0$ , then the ellipsoid  $\mathcal{E}_P = \{x | x^T P x \leq 1\}$  is a stable domain for system (15) (every trajectory initiating in  $\mathcal{E}_P$  goes to zero).

With the above basic principle in mind, we see that to guarantee our design specifications, it suffices to write a robust stability (or more generally, robust performance) condition for an associated uncertain system, associated with a condition that guarantees the state to be bounded. For more details, we refer to Reference 1.

In view of assumption A.2, we can always assume that  $|x_i| \leq x_{i,\max}$  implies some uniform bound on the matrix  $\tilde{\Delta}(x, t)$ . To simplify, we assume that this bound writes  $\|\tilde{\Delta}(x, t)\| \leq 1$ . (We can always use loop transformations to end up with such a bound.)

The robust performance condition, derived from inequality (9), is that there exists a Lyapunov function matrix  $\tilde{P}$ , and a scaling matrix  $\tilde{S}$  commuting with any  $\tilde{\Delta}$  of the form (6), such that

$$\begin{bmatrix} 2\alpha\tilde{P} + \tilde{A}^T\tilde{P} + \tilde{P}\tilde{A} + \tilde{C}_q^T\tilde{S}\tilde{C}_q & \tilde{P}\tilde{B}_p + \tilde{C}_z^T\tilde{D}_{zp} + \tilde{C}_q^T\tilde{S}\tilde{D}_{qp} & \tilde{P}\tilde{B}_w + \tilde{C}_q^T\tilde{S}\tilde{D}_{qw} \\ * & \tilde{D}_{zp}^T\tilde{D}_{zp} + \tilde{D}_{qp}^T\tilde{S}\tilde{D}_{qp} - \tilde{S} & \tilde{D}_{qp}^T\tilde{S}\tilde{D}_{qw} \\ * & * & \tilde{D}_{qw}^T\tilde{S}\tilde{D}_{qw} - \gamma^2 I \end{bmatrix} < 0 \tag{16}$$

We note that the above inequality also guarantees well-posedness (specification S.1). The bound  $V(\tilde{x}(0)) \leq \lambda$  for every  $x(0) \in \mathcal{P}(x_0)$  can be written as

$$\begin{bmatrix} v_j \\ 0 \end{bmatrix}^T \tilde{P} \begin{bmatrix} v_j \\ 0 \end{bmatrix} \leq \lambda, \quad j = 1, \dots, L$$

With the above two conditions in force, we know that for every trajectory in  $\mathcal{X}(x_0, w_{\max})$ , inequality (12) holds. Specification S.3 then holds if

$$\|[O \ \tilde{C}_u]\tilde{x}\|_2 \leq u_{\max} \quad \text{for every } \tilde{x}, \tilde{x}^T \tilde{P} \tilde{x} \leq (\lambda + \gamma^2 w_{\max}^2)$$

We rewrite the above condition as follows:

$$[O \ \bar{C}_u] \tilde{P}^{-1} [O \ \bar{C}_u]^T \leq \mu \mu_{\max}^2 I \quad \text{where } \mu(\lambda + \gamma^2 w_{\max}^2) = 1$$

Likewise, the output peak bound (specification S.4) then holds if

$$\begin{bmatrix} e_j \\ 0 \end{bmatrix}^T \tilde{P}^{-1} \begin{bmatrix} e_j \\ 0 \end{bmatrix} \leq \mu x_{j,\max}^2, \quad j = 1, \dots, n$$

where  $e_j$  is the  $j$ th column of the  $n \times n$  identity matrix.

4.4. Matrix inequality synthesis conditions

As seen in References 1 and 2, it is possible to formulate the synthesis conditions in the form of LMI conditions associated with a non-convex constraint. These inequalities depend, in particular, on  $P$  and  $Q$ , the upper-left  $n \times n$  blocks of  $\tilde{P}$  and  $\tilde{P}^{-1}$ , respectively.

Let  $\mathcal{N}$  be a matrix whose columns span the null space of  $[C_y \ D_{yp} \ D_{yw}]^T$ . The synthesis conditions are

$$\exists P, Q, S = \begin{bmatrix} S_{\text{meas}} & 0 \\ 0 & S_{\text{unk}} \end{bmatrix}, \quad T = \begin{bmatrix} T_{\text{meas}} & 0 \\ 0 & T_{\text{unk}} \end{bmatrix} \in \mathcal{S}(r)$$

$\lambda, \mu > 0$  and  $Y$  such that

$$\begin{bmatrix} P & I \\ I & Q \end{bmatrix} \geq 0, \quad \begin{bmatrix} S & I \\ I & T \end{bmatrix} \geq 0 \tag{17}$$

$$\mathcal{N}^T \begin{bmatrix} A^T P + PA + C_q^T S C_q & PB_p + C_q^T S D_{qp} & PB_w + C_q^T S D_{qw} \\ + 2\alpha P + C_z^T C_z & + C_z^T D_{zp} & \\ * & D_{qp}^T S D_{qp} - S & D_{qp}^T S D_{qw} \\ * & + D_{zp}^T D_{zp} & \\ * & * & D_{qw}^T S D_{qw} - \gamma^2 I \end{bmatrix} \mathcal{N} < 0 \tag{18}$$

$$\begin{bmatrix} AQ + QA^T + B_u Y + Y^T B_u & QC_q^T + Y^T D_{qu}^T & QC_z^T + Y^T D_{zu}^T \\ + 2\alpha Q + B_p T B_p^T + \gamma^{-2} B_w B_w^T & + B_p T D_{qp}^T + \gamma^{-2} B_w D_{qw}^T & B_p T D_{zp}^T \\ * & D_{qp} T D_{qp}^T - T & D_{qp} T D_{zp}^T \\ * & + \gamma^{-2} D_{qw} D_{qw}^T & \\ * & * & D_{zp} T D_{zp}^T - I \end{bmatrix} < 0 \tag{19}$$

$$v_i^T P v_i \leq \lambda, \quad i = 1, \dots, L \tag{20}$$

$$\begin{bmatrix} \mu \mu_{\max}^2 I & Y & 0 \\ Y^T & Q & I \\ 0 & I & P \end{bmatrix} \geq 0 \tag{21}$$

$$\mu x_{j,\max}^2 - e_j^T Q e_j \geq 0, \quad j = 1, \dots, n \tag{22}$$

and

$$S_{\text{unk}} T_{\text{unk}} = I_{v_{\text{unk}}}, \quad \mu(\lambda + w_{\text{max}}^2 \gamma^2) = 1 \tag{23}$$

4.5. Solving for the conditions

Every condition above is an LMI, except for the non-convex equations (23). When (17) holds, enforcing this condition can be done by imposing  $\text{Tr } S_{\text{unk}} T_{\text{unk}} + \mu(\lambda + w_{\text{max}}^2 \gamma^2) = v_{\text{unk}} + 1$  with the additional LMI constraint

$$\begin{bmatrix} \mu & I \\ I & \lambda + \gamma^2 w_{\text{max}}^2 \end{bmatrix} \geq 0 \tag{24}$$

In fact, the problem belongs to the class of ‘cone-complementarity problems’, which are based on LMI constraints of the form

$$F(V, W, Z) \geq 0, \quad \begin{bmatrix} V & I \\ I & W \end{bmatrix} \geq 0 \tag{25}$$

where  $V, W, Z$  are matrix variables ( $V$  and  $W$  being symmetric and of same size), and  $F(V, W, Z)$  is a matrix-valued, affine function, with  $F$  symmetric. The corresponding cone-complementarity problem is

$$\text{minimize } \text{Tr } VW \text{ subject to (25).}$$

The heuristic proposed in Reference 28, which is based on solving a sequence of LMI problems, can be used to solve the above problem. This heuristic is guaranteed to converge to a stationary point.

*Cone complementarity algorithm  $\mathcal{H}$ :*

1. Find  $V, W$  and  $Z$  that satisfy the LMI constraints (25). If the problem is infeasible, stop. Otherwise, set  $k = 1$ .
2. Find  $V_k, W_k$  that solve the LMI problem

$$\text{minimize } \text{Tr}(V_{k-1}W + W_{k-1}V) \text{ subject to (25).}$$

3. If the objective  $\text{Tr}(V_{k-1}W_k + W_{k-1}V_k)$  has reached a stationary point, stop. Otherwise, set  $k = k + 1$  and go to (2).

Depending on the number of measured and unmeasured parameters (related to the size of matrices  $\Delta_{\text{meas}}$  and  $\Delta_{\text{unk}}$ , respectively,  $v_{\text{meas}}$  and  $v_{\text{unk}}$ , we recover the previous results:

- If  $v_{\text{unk}} = 0$  (there are no unmeasured nonlinearities), our conditions are convex, and equivalent to those obtained by Packard<sup>26</sup> for gain-scheduling of LPVs. In this case, there is no need to use the heuristic above.
- If  $v_{\text{meas}} = 0$  (there are no measured nonlinearities), our conditions are equivalent to the conditions obtained in robust synthesis based on quadratic stability.<sup>24</sup>

We note that, in the state-feedback case, the heuristic is not necessary and the problem is convex. Note also that, in this case, the controller is linear. This is basically due to the use of quadratic Lyapunov function, which implies a separation principle; in the output-feedback case,

the controller can be interpreted as a linear feedback  $u = K\hat{x}$ , where  $\hat{x}$  is the state estimated by a nonlinear observer.<sup>7</sup>

#### 4.6. Controller reconstruction

When the algorithm exits successfully, i.e. when  $S_{\text{unk}}T_{\text{unk}} \simeq I$  and  $\mu(\lambda + w_{\text{max}}^2\gamma^2) \simeq 1$  (note that a stopping criterion is proposed in Reference 28), then we can reconstruct an appropriate controller as follows.

First, we reconstruct a Lyapunov function for the closed-loop system  $V(\tilde{x}) = \tilde{x}^T \tilde{P} \tilde{x}$ , where  $\tilde{P}$  is computed directly from the variables  $P$  and  $Q$ . When  $P > Q^{-1}$ , the appropriate closed-loop Lyapunov functions are parameterized by an arbitrary invertible matrix  $M \in \mathbb{R}^{n \times n}$ , via the formula

$$\tilde{P} = \begin{bmatrix} I & 0 \\ 0 & M^T \end{bmatrix} \begin{bmatrix} P & I \\ I & (P - Q^{-1})^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix} \quad (26)$$

(The choice of  $M$  is irrelevant: it corresponds to a change of coordinates of the controller's states,  $\tilde{x} \rightarrow M\tilde{x}$ .) Similar formulae can be used to reconstruct an appropriate scaling matrix  $\tilde{S}$  from the variables  $S$  and  $T$ . We can also compute a controller matrix  $\tilde{C}_u$  from the variables  $Y, P, Q$ , setting  $\tilde{C}_u = YN^{-T}$ , where  $N$  is the upper-right  $n \times n$  block of  $\tilde{P}^{-1}$ .

Once a Lyapunov function  $\tilde{P}$  and a scaling matrix  $\tilde{S}$  is constructed, we proceed to reconstruct the controller matrix  $K$ , as follows. We note that the controller gain matrix  $K$  (defined in Section 3.3) appears linearly in the analysis condition (16) when  $\tilde{P}$  and  $\tilde{S}$  are fixed. Therefore, we can compute  $K$  by solving an LMI feasibility problem. Note that analytic controller formulae are given in Reference 32.

#### 4.7. Overview of the design method

Our design method can be summarized as follows.

1. Describe the design problem mathematically, as in Section 3.4. Choose a set of design parameters  $\alpha, u_{\text{max}}, x_{\text{max}}$ , etc.
2. Form an LFR of the open-loop system, and normalize it (use the design parameter  $x_{\text{max}}$ ).
3. Solve the synthesis conditions using algorithm  $\mathcal{H}$ .
4. If the design is satisfactory, reconstruct the controller and exit.
5. Relax the design parameters and go to step 2.

We note that the method requires the parameters  $x_{i,\text{max}}$  to be finite if the corresponding state coordinate  $x_i$  appears in the nonlinear element  $\Delta(x)$ .

The above method relies on an appropriate choice of the design parameters  $\alpha, u_{\text{max}}, x_{\text{max}}$ , etc. This choice must be made on a heuristic basis. However, we stress that the designer has several guidelines for this choice.

- Every parameter has an obvious physical interpretation, which helps deciding whether to relax, or strengthen, a given specification.
- The approach enables to compute trade-off curves. For example, we can compute the *optimal*  $u_{\text{max}}$  attainable via the method, as a function of the decay rate  $\alpha$  (the other parameters being fixed).

5. NUMERICAL RESULTS

The numerical results were obtained using the public domain toolbox MRCT<sup>29,30</sup> built on top of the software lmitool<sup>31</sup> and the semidefinite programming package SP.<sup>33</sup>

5.1. The benchmark problem

Consider the unbalanced oscillator described in Figure 3. The RTAC is built with a cart (weigh  $M$ ) fixed to the wall by a linear spring (stiffness  $k$ ) and constrained to have one-dimensional travel along the  $Z$ -axis. An embedded proof-mass actuator (mass  $m$  and moment of inertia  $I$ ) is attached to the centre of mass of the cart and can rotate in the horizontal plane. The radius of rotation is  $e$ . The cart is submitted to a disturbance  $F$  and a control torque  $N$  is applied to the proof mass.

5.2. Nonlinear equations of motion

Let  $\theta$  and  $\dot{\theta}$  denote the angular position and velocity of the proof mass, and let  $Z$  and  $\dot{Z}$  denote the translational position and velocity of the cart. The nonlinear equations of motion are

$$\begin{aligned} (M + m)\ddot{Z} + me\ddot{\theta}\cos\theta &= me\dot{\theta}^2\sin\theta - kZ + F \\ me\ddot{Z}\cos\theta + (I + me^2)\ddot{\theta} &= N \end{aligned} \tag{27}$$

Define

$$\begin{aligned} \xi &= \sqrt{\frac{M + m}{I + me^2}}Z, \quad \tau = \sqrt{\frac{k}{M + m}}t \\ u &= \frac{M + m}{k(I + me^2)}N, \quad w = \frac{1}{k}\sqrt{\frac{M + m}{I + me^2}}F, \quad \varepsilon = \frac{me}{\sqrt{(I + me^2)(M + m)}} \end{aligned} \tag{28}$$

The normalized nonlinear equations of motion of the cart are

$$\begin{aligned} \ddot{\xi} + \varepsilon\ddot{\theta}\cos\theta &= \varepsilon\dot{\theta}^2\sin\theta - \xi + w \\ \varepsilon\dot{\xi}\cos\theta + \ddot{\theta} &= u \end{aligned} \tag{29}$$

The above equations can be written as

$$\dot{x} = f(x) + g(x)u + d(x)w$$

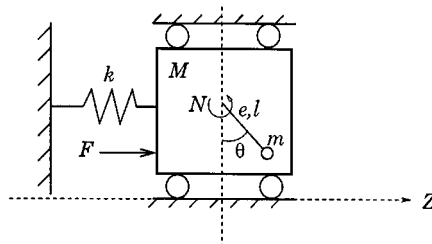


Figure 3. Rotational actuator to control a translational oscillator

where the state is  $x = [\xi \ \dot{\xi} \ \theta \ \dot{\theta}]^T$ , and

$$f(x) = \begin{bmatrix} x_2 \\ \frac{-x_1 + \varepsilon x_4^2 \sin x_3}{1 - \varepsilon^2 \cos^2 x_3} \\ x_4 \\ \frac{\varepsilon \cos x_3 (x_1 - \varepsilon x_4^2 \sin x_3)}{1 - \varepsilon^2 \cos^2 x_3} \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ \frac{-\varepsilon \cos x_3}{1 - \varepsilon^2 \cos^2 x_3} \\ 0 \\ \frac{1}{1 - \varepsilon^2 \cos^2 x_3} \end{bmatrix}, \quad d(x) = \begin{bmatrix} 0 \\ 1 \\ \frac{1}{1 - \varepsilon^2 \cos^2 x_3} \\ 0 \\ \frac{-\varepsilon \cos x_3}{1 - \varepsilon^2 \cos^2 x_3} \end{bmatrix}$$

The measured outputs are  $\xi$  and  $\theta$  so that

$$y^T = [\xi \ \theta]$$

Motivated by Biotras *et al.*,<sup>8</sup> we choose the controlled output to be

$$z^T = [\sqrt{0.1}\xi \ \sqrt{0.1}\theta \ u]$$

### 5.3. LFR model

In order to use the method, we need to work out an LFR model for the open-loop system. To this end, we choose a (state-dependent) ‘parameter vector’  $\delta$ , such that the state-space equations of the system are linear when this parameter is frozen.

In the present case, we may choose  $\delta = [\delta_1 \ \delta_2]$ , with

$$\delta_1 = \cos x_3 \quad \text{and} \quad \delta_2 = x_4 \sin x_3$$

This choice leads to state-space equations of the form (1), with

$$\mathbf{A}(\delta) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{1}{1 - \varepsilon^2 \delta_1^2} & 0 & 0 & \frac{\varepsilon \delta_2}{1 - \varepsilon^2 \delta_1^2} \\ 0 & 0 & 0 & 1 \\ \frac{\varepsilon \delta_1}{1 - \varepsilon^2 \delta_1^2} & 0 & 0 & -\frac{\varepsilon^2 \delta_1 \delta_2}{1 - \varepsilon^2 \delta_1^2} \end{bmatrix}$$

$$\mathbf{B}_u(\delta) = \begin{bmatrix} 0 \\ -\frac{\varepsilon \delta_1}{1 - \varepsilon^2 \delta_1^2} \\ 0 \\ \frac{1}{1 - \varepsilon^2 \delta_1^2} \end{bmatrix} \quad \text{and} \quad \mathbf{B}_w(\delta) = \begin{bmatrix} 0 \\ 1 \\ 0 \\ -\frac{\varepsilon \delta_1}{1 - \varepsilon^2 \delta_1^2} \end{bmatrix}$$

The next task is to form an LFR for the parameter-dependent matrix  $\mathbf{M}(\delta) = [\mathbf{A}(\delta) \ \mathbf{B}_u(\delta) \ \mathbf{B}_w(\delta)]$ . This task can be done using construction rules detailed in Appendix A. The basic idea of this construction is to decompose the matrix  $\mathbf{M}$  into sums of fractions where the

parameters appear with degree one. We first obtain

$$\begin{bmatrix} \mathbf{A}(\delta)^T \\ \mathbf{B}_u(\delta)^T \\ \mathbf{B}_w(\delta)^T \end{bmatrix}^T = \left( \begin{bmatrix} 0 \\ -\frac{1}{2} \\ 0 \\ -\frac{1}{2} \end{bmatrix} \frac{1}{1 + \varepsilon\delta_1} + \begin{bmatrix} 0 \\ -\frac{1}{2} \\ 0 \\ \frac{1}{2} \end{bmatrix} \frac{1}{1 - \varepsilon\delta_1} \right) \begin{bmatrix} 1 \\ 0 \\ 0 \\ -\varepsilon\delta_2 \\ \frac{\varepsilon\delta_1}{-1} \end{bmatrix}^T + \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

We note that the parameter  $\delta_1$  appears three times (with degree one) in the above expression, while  $\delta_2$  appears only once. Therefore, the matrix  $\Delta$  (appearing in (2)) is

$$\Delta = \mathbf{diag}(\delta_1 I_3, \delta_2)$$

With the above representation, we may rewrite the state-space equations of the RTAC as (2), where the non-zero matrices are

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad B_u = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad C_q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B_w = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$B_p = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{\varepsilon}{2} & -\frac{\varepsilon}{2} & -\varepsilon & \varepsilon \\ 0 & 0 & 0 & 0 \\ \frac{\varepsilon}{2} & \frac{\varepsilon}{2} & 0 & 0 \end{bmatrix}, \quad D_{qu} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad D_{qp} = \begin{bmatrix} -\varepsilon & 0 & \varepsilon & -\varepsilon \\ 0 & \varepsilon & \varepsilon & -\varepsilon \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad D_{qw} = \begin{bmatrix} -1 \\ -1 \\ 0 \\ 0 \end{bmatrix}$$

$$C_z = \sqrt{0.1} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad D_{zu} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad C_y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

5.4. Control design specifications

Throughout this section, we have set the design parameter  $x_{\max}$  to

$$x_{\max} = [1.28 \ \infty \ 0.5 \ 0.5]^T$$

We note that the value of  $\xi_{\max} = 1.28$  is imposed by the physical configuration of the system,<sup>3</sup> while the other output bounds concern the maximal angular values,  $\theta_{\max}$  and  $\dot{\theta}_{\max}$ . Note that we do not impose a bound on the state  $\dot{\xi}$ . This is allowed, since the latter variable appears linearly in the equations of motion.

5.5. A mixed control design

We have designed a state-feedback and an output-feedback controller for the system, based on the mixed control design strategy described in Section 4.2.



We first seek a state-feedback controller for our system. Our design parameters are

$$\alpha = 0.02, \quad \gamma = 10, \quad x_0 = [0.5 \ 0 \ 0 \ 0]^T \quad (30)$$

Since the problem is convex in this case, we have taken the additional degree of freedom to minimize the upper bound on command input  $u_{\max}$ , subject to the remaining LMI constraint.

We have found the guaranteed bound  $u_{\max} = 0.4586$ , and the resulting state-feedback, static linear controller is  $u = \bar{C}_u x$ , where

$$\bar{C}_u = 0.1 [1.20 \ 0.76 \ -6.49 \ -7.05]$$

An output-feedback controller was synthesized, based on the same design parameters (30), but with a relaxation process on  $u_{\max}$ ,  $x_{\max}$ ,  $\alpha$  and  $\gamma$ . The controller is given by the LFR (4), with gain matrices

$$\bar{A} = \begin{bmatrix} -2.82 & -1.46 & 0.13 & -100.81 \\ -5.20 & -6.84 & -2.13 & 73.68 \\ -2.97 & -4.51 & -1.91 & -2.66 \\ 9.57 & -10.40 & 12.28 & -1166.03 \end{bmatrix}, \quad \bar{B}_y = \begin{bmatrix} 22.22 & -1042.92 \\ -2.32 & -1977.47 \\ -29.92 & -1318.11 \\ -2236.06 & 1.34 \end{bmatrix}$$

$$\bar{B}_p = 0.01 \begin{bmatrix} -6.10 & -247.25 & -119.77 \\ 4.48 & 181.32 & 86.11 \\ -0.21 & -6.95 & 5.64 \\ -11.87 & -481.25 & -233.35 \end{bmatrix}, \quad \bar{C}_u = 0.01 \begin{bmatrix} 0.04 \\ 0.63 \\ 0.33 \\ -46.43 \end{bmatrix}$$

$$\bar{C}_q = 0.01 \begin{bmatrix} 0.02 & 0.38 & 0.20 & -27.53 \\ 0.19 & 3.00 & 1.59 & -219.72 \\ -1.32 & -7.80 & -2.73 & 486.07 \end{bmatrix}$$

$$\bar{D}_{qy} = 0.01 \begin{bmatrix} -0.04 & -0.18 \\ -0.74 & -1.97 \\ -229.26 & -379.94 \end{bmatrix}, \quad \bar{D}_{qp} = 0.01 \begin{bmatrix} 0.03 & 1.29 & 0.61 \\ -0.14 & -5.46 & -2.58 \\ 0.29 & 11.88 & 5.90 \end{bmatrix}$$

This control law has been computed on a Sparc 4 workstation. It required 20 s of CPU time and 977 Mflops for the solution of the LMI conditions, and 17 s of CPU time and 3.05 Gflops for the controller reconstruction phase. It only required one iteration in the algorithm  $\mathcal{H}$ .

We show in Figure 4 the Bode plots of the nominal controller with inputs  $\theta$  and  $\zeta$ , and with output the command  $u$ .

Figure 5 describes the closed-loop behaviour in response to zero external disturbances ( $F(t) \equiv 0$ ) and a non-zero initial condition for both controllers. These plots show the actual quantities (without the normalization (28)), i.e.  $N$  in Nm,  $w$  in N,  $Z$  in m,  $\theta$  in rad,  $t$  in s. We can note that the various bounds that were required are ensured, and that the settling time (around 7 s for the worst-case initial condition) is satisfactory.

In Figure 6, we show the behaviour of the closed-loop system (with the output-feedback controller), in response to an external disturbance  $F(t) = 1.8 \sin 10\pi t$ . The plots demonstrate how

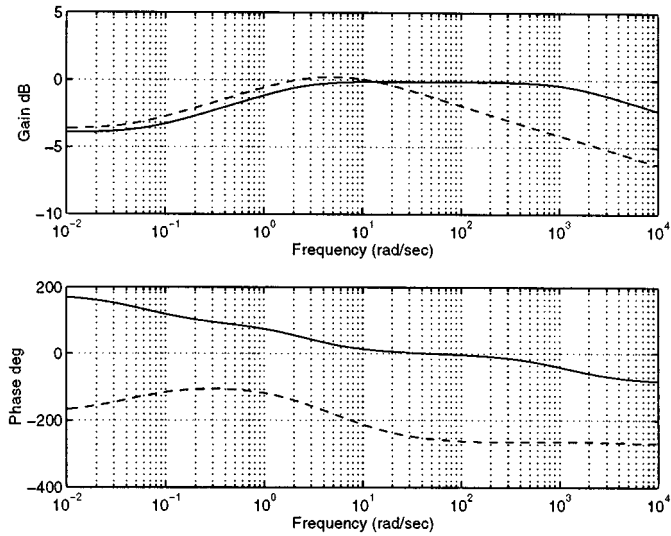


Figure 4. Bode plots of the nominal output-feedback controller, with input  $\xi$  in plain line and with input  $\theta$  in dashed line

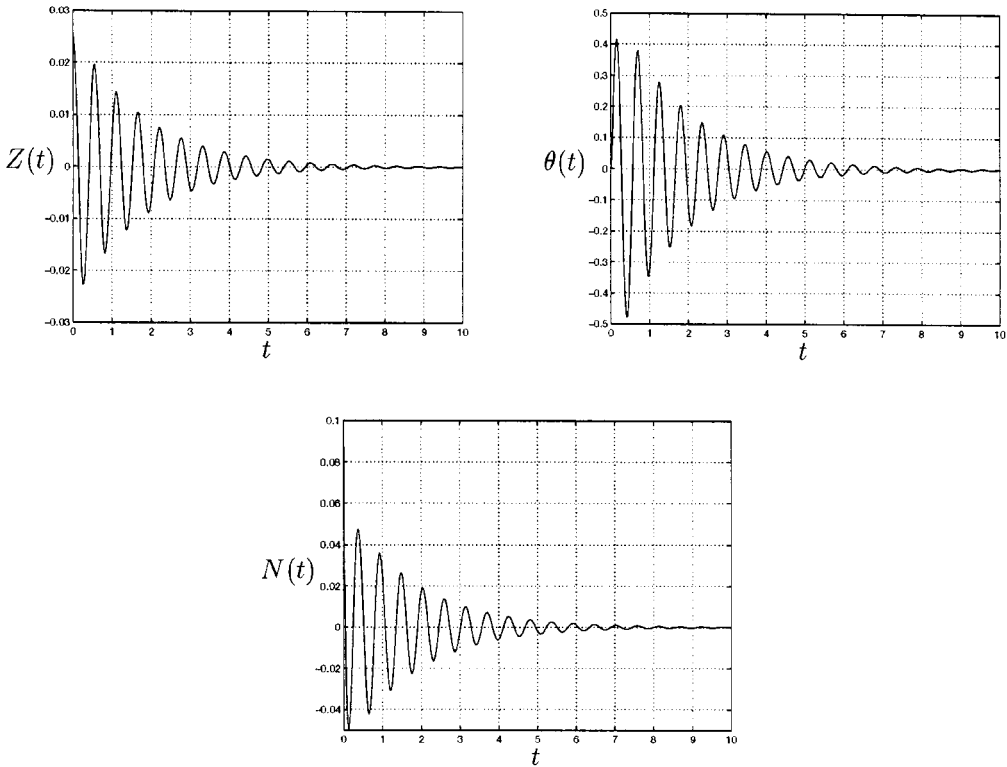


Figure 5. Closed-loop system time history with  $F(t) \equiv 0$ , zero initial conditions except for  $Z(0) = 0.023$  m, for the output-feedback controller given in Section 5.5.  $t$  is in s,  $Z$  in m,  $\theta$  in rad and  $N$  in N

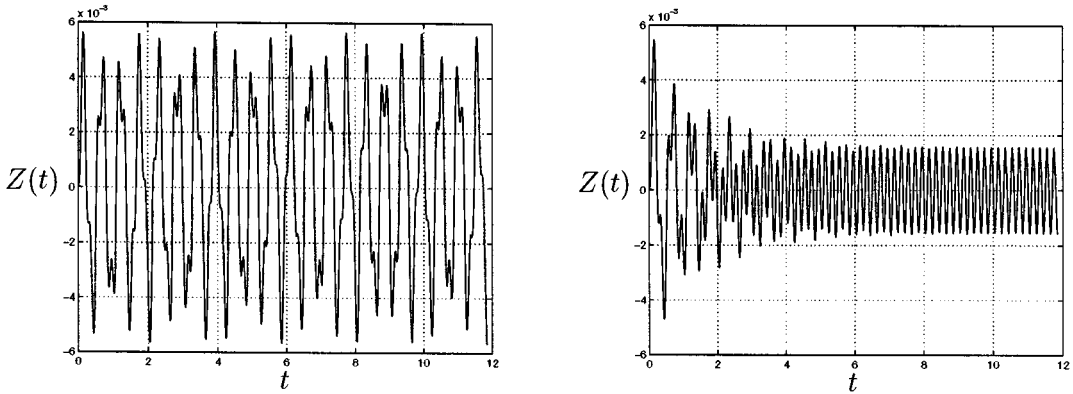


Figure 6. Open-loop (on the left) and closed-loop (on the right) system time history subject to zero initial condition and a non-zero external disturbance  $F(t) = 1.8 \sin 10\pi t$ , for the output-feedback controller given in Section 5.5.  $t$  is in s,  $Z$  in m

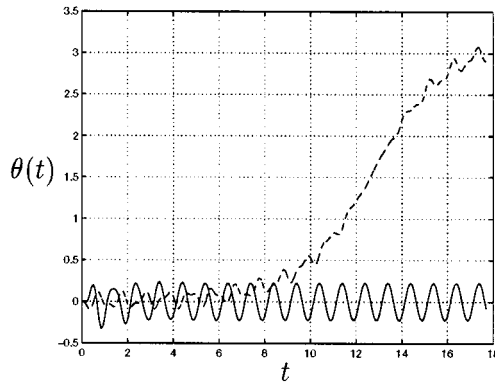


Figure 7. Open-loop (in dashed line) and closed-loop (in plain line) system time history subject to zero initial condition and a non-zero external disturbance  $F(t) = \sin 2\pi t$ , for the output-feedback controller given in Section 5.5.  $t$  is in s,  $\theta$  in rad

the system rejects the signal  $F(t)$ . We also show the corresponding open-loop responses, which turns out to be stable.

In Figure 7, we choose another disturbance signal  $F(t) = \sin 2\pi t$ , that has a much lower frequency than the previous one. This time, the corresponding open-loop behaviour is unstable. In contrast, the controlled behaviour is still stable.

5.6. Some trade-off curves

It is interesting to compute predicted trade-off curves. Figure 8 shows trade-off curves decay rate/control effort, with  $\gamma = \infty$  (right-hand side) and performance/control effort, with  $\alpha = 0$  (left-hand side). The design is based on the vector  $x_0$  given by (30). Performance is measured by  $\gamma$ , the upper bound on the closed-loop  $\mathcal{L}_2$ -gain and control effort is measured by the upper bound on peak command input  $N_{max}$ . In each plot, two curves are shown: one for state feedback (in plain line) and the other for output feedback (in dashed line). Figure 9 shows similar results for the basic

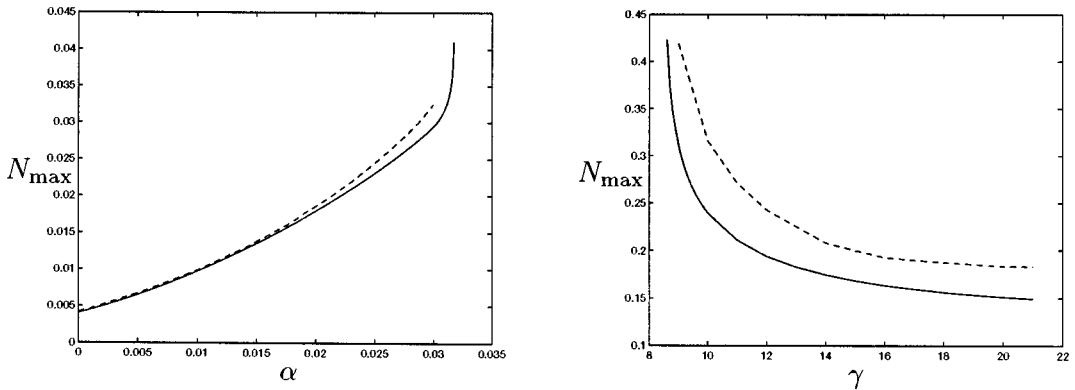


Figure 8. Trade-off plots command input/decay rate for the state- (plain line) the output-feedback (dashed line) controller. On the right side,  $\gamma = \infty$ , and on the left side,  $\alpha = 0$ .  $N_{\max}$  is in N

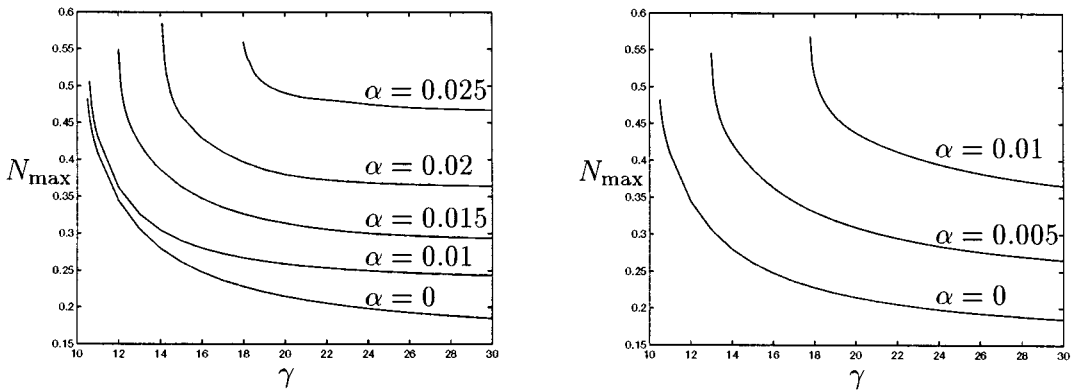


Figure 9. Trade-off plots command input/ $\mathcal{L}_2$ -performance for the basic (on the left) and mixed (on the right) strategies as defined in Section 4.2, using state-feedback control.  $N_{\max}$  is in N

and mixed strategy, using state-feedback control. Clearly, the basic strategy is more conservative than the mixed one (note that both strategies coincide when  $\alpha = 0$ ).

5.7. On other approaches to the RTAC problem

It is hard to compare our results with those obtained by other methods for the RTAC problem, since the system parameters vary from one paper to the other. For instance, the important coupling parameter  $\varepsilon = 0.2$  in the required specifications, as well as in our paper. In References 5, 6 and 34,  $\varepsilon = 0.1$  and in Reference 8,  $\varepsilon = 0.5$ . Moreover, the units in the plots (especially for the time scale) are sometimes unclear in some of the above papers (recall there is a factor 10 between the ‘real time’  $t$  and the normalized time  $\tau$ ). With these differences taken into account, the performances of our controller, in terms of settling time and disturbance rejection, appear to be equivalent to those presented in References 5, 7 and 8. Such a result is very satisfactory, in view of the fact that the method is systematic and handles multicriteria problems.

## 6. CONCLUSIONS

In this paper, we showed some key aspects of a methodology for nonlinear control introduced in Reference 1 and applied it to the RTAC benchmark problem. The approach can be adapted to a large class of nonlinear systems, and also to *uncertain* nonlinear systems,<sup>2</sup> in particular, it is not restricted to systems with bounded nonlinearities. It is very systematic and handles *multicriteria* control. One important aspect of the method is that it allows computing trade-off curves for multicriteria design.

Recently, accurate robustness analysis tools based on multiplier theory have been devised, see, e.g. Megretsky and Rantzer<sup>17</sup> and Balakrishnan.<sup>18</sup> A complete methodology for nonlinear design should include these tools for controller validation.

## ACKNOWLEDGEMENTS

The authors wish to thank the Associate Editor and the anonymous reviewers for their precious help in improving the manuscript.

## REFERENCES

1. El Ghaoui, L. and G. Scorletti, 'Control of rational systems using linear-fractional representations and linear matrix inequalities', *Automatica*, **32**, 1273–1284 (1996).
2. Dussy, S. and L. El Ghaoui, Multiobjective bounded control of uncertain nonlinear systems: an inverted pendulum example, in *Control of Uncertain Systems with Bounded Inputs*, Springer, Berlin, 1997, pp. 55–73.
3. Bupp, R., D. Bernstein and V. Coppola, 'A benchmark problem for nonlinear control design: problem state, experimental testbed, and passive nonlinear compensation', *Proc. IEEE American Control Conf.*, Seattle, WA, June 1995, pp. 4363–4367.
4. Abed, E., Y.-S. Chou, A. Guran and A. Tits, 'Nonlinear stabilization and parametric optimization in the benchmark nonlinear control design problem', *Proc. IEEE American Control Conf.*, Seattle, WA, June 1995, pp. 4357–4359.
5. Wan, C.-J., D. Bernstein and V. Coppola, 'Global stabilization of the oscillating eccentric rotor', *Proc. IEEE Conf. Decision and Control*, Orlando, FL, December 1994, pp. 4024–4029.
6. Bupp, R., C.-J. Wan, V. Coppola and D. Bernstein, 'Design of a rotational actuator for global stabilization of translational motion', *Proc. ASME Winter Meeting*, Vol. 75, Chicago, IL, 1994, pp. 449–456.
7. Kanallakopoulos, I. and J. Zhao, 'Tracking and disturbance rejection for the benchmark nonlinear control problem', *Proc. IEEE American Control Conf.*, Seattle, WA, June 1995, pp. 4360–4362.
8. Tsiotras, P., M. Corless and M. Rotea, 'An  $\mathcal{L}_2$  disturbance attenuation approach to the nonlinear benchmark problem', *Proc. IEEE American Control Conf.*, Seattle, WA, June 1995, pp. 4352–4356.
9. Monopoli, R., 'Synthesis techniques employing the direct method', *IEEE Trans. Automat. Control*, **10**, 369–370 (1965).
10. Pomet, J.-B., R. Hirschorn and W. Cebuhar, 'Dynamic output feedback regulation for a class of nonlinear systems', *Math. Control Signals Systems* **6**, 106–124 (1993).
11. Isidori, A., *Nonlinear Control Systems: An Introduction*, 2nd edn, Springer, Berlin, 1993.
12. Van der Schaft, A., ' $\mathcal{L}_2$ -gain analysis of nonlinear systems and nonlinear state-feedback  $\mathcal{H}_\infty$  control', *IEEE Trans. Automat. Control* **37**, 770–784 (1992).
13. Isidori, A., ' $\mathcal{H}_\infty$  control via measurement feedback for affine nonlinear systems', *Int. J. Robust Nonlinear Control*, **4**, 553–574 (1994).
14. Lu, W. and J. Doyle, ' $\mathcal{H}_\infty$  control of nonlinear systems: a convex characterization', *IEEE Trans. Automat. Control*, **40**, 1668–1675 (1995).
15. Gahinet, P., P. Apkarian and M. Chilali, 'Affine parameter-dependent Lyapunov functions for real parametric uncertainty', *Proc. IEEE Conf. Decision and Control*, Lake Buena Vista, FL, December 1994, pp. 2026–2031.
16. Feron, E., P. Apkarian and P. Gahinet, ' $\mathcal{S}$ -procedure for the analysis of control systems with parametric uncertainties via parameter-dependent Lyapunov functions', *Proc. IEEE American Control Conf.*, Seattle, WA, June 1995, pp. 968–972.
17. Megretski, A. and A. Rantzer, 'System analysis via integral quadratic constraints', *Proc. IEEE Conf. Decision and Control*, Orlando, FL, December 1994, pp. 3062–3067.
18. Balakrishnan, V., 'Linear matrix inequalities in robustness analysis with multipliers', *Systems Control Lett.*, **25**, 265–272 (1995).

19. Haddad, W. and D. Bernstein, 'Parameter-dependent Lyapunov functions and the Popov criterion in robust analysis and synthesis', *IEEE Trans. Automat. Control*, **40**, 536–543 (1995).
20. Sparks, A. and D. Bernstein, 'Robust controller synthesis using scaled Popov bounds with uncertain  $A$ ,  $B$  and  $C$  matrices', *IFAC 13th Triennial World Congress*, San Francisco, LA, 1996, pp. 333–338.
21. Haddad, W., V. Kapila and D. Bernstein, 'Robust  $\mathcal{H}_\infty$  stabilization via parametrized Lyapunov bounds', *IEEE Trans. Automat. Control*, **42**(2), 243–248 (1997).
22. Packard, A., K. Zhou, P. Pandey and G. Becker, 'A collection of robust control problems leading to LMI's', *Proc. IEEE Conf. Decision and Control*, Brighton, England, December 1991, pp. 1245–1250.
23. Lu, W. and J. Doyle, ' $\mathcal{H}_\infty$  control of LFT systems: an LMI approach', *Proc. IEEE Conf. Decision and Control*, Tucson, AZ, December 1992, pp. 1997–2001.
24. Doyle, J., A. Packard and K. Zhou, 'Review of LFTs, LMIs and  $\mu$ ', *Proc. IEEE Conf. Decision and Control*, Brighton, England, December 1991, pp. 1227–1232.
25. Boyd, S., L. El Ghaoui, E. Feron and V. Balakrishnan, *Linear Matrix Inequality in Systems and Control Theory*, SIAM, Philadelphia, PA, 1994.
26. Packard, A., 'Gain scheduling via linear-fractional transformation', *Systems Control Lett.*, **22**, 79–92 (1994).
27. Becker, G. and A. Packard, 'Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback', *Systems Control Lett.*, **23**, 205–215 (1994).
28. El Ghaoui, L., F. Oustry and M. Ait Rami, 'A cone complementary linearization algorithm for static output-feedback and related problems', *IEEE Trans. Automat. Control*, (1997).
29. Dussy, S. and L. El Ghaoui, 'Multiobjective robust control toolbox for LMI-based control', *Proc. IFAC Symp. Computer Aided Control Systems Design*, Gent, Belgium, April 1997, pp. 353–358.
30. Dussy, S. and L. El Ghaoui, *Multiobjective Robust Control Toolbox (MRCT): User's Guide*, 1996. Available via <http://www.ensta.fr/~gropco/staff/dussy/gocpage.html>.
31. El Ghaoui, L., R. Nikoukhah and F. Delebecque, 'LMI TOOL: A Front-End for LMI Optimization, User's Guide', February 1995. Available via anonymous ftp to <ftp://ensta.fr> under `/pub/elghaoui/lmitool`.
32. Iwasaki, T. and R. Skelton, 'All controllers for the general  $\mathcal{H}_\infty$  control problems: LMI existence conditions and state space formulas', *Automatica*, **30**(8), 1307–1317 (1994).
33. Vandenberghe, L. and S. Boyd, *SP, Software for Semidefinite Programming, User's Guide*, December 1994. Available via anonymous ftp to [isl.stanford.edu](http://isl.stanford.edu) under `/pub/boyd/semidef_prog`.
34. Kinsey, R., D. Mingori and R. Rand, 'Limited torque spinup of an unbalanced rotor on an elastic support', *Proc. IEEE American Control Conf.*, Seattle, WA, June 1995, pp. 4368–4373.

## APPENDIX A. LFR CONSTRUCTION

In this section, we provide some details about how an LFR model such as (2) can be constructed, for a parameter-dependent system (1).

Our framework includes the case when parameters perturb each coefficient of the state-space matrices in a (polynomial or) rational manner. This is due to the representation lemma given below.

### *Theorem A.1*

For any rational matrix function  $\mathbf{M}: \mathbb{R}^k \rightarrow \mathbb{R}^{n \times c}$ , with no singularities at the origin, there exist nonnegative integers  $r_1, \dots, r_p$ , and matrices  $M \in \mathbb{R}^{n \times c}$ ,  $L \in \mathbb{R}^{n \times N}$ ,  $R \in \mathbb{R}^{N \times c}$ ,  $D \in \mathbb{R}^{N \times N}$ , with  $N = r_1 + \dots + r_p$ , such that  $\mathbf{M}$  has the following LFR: for all  $\delta$  where  $\mathbf{M}$  is defined,

$$\mathbf{M}(\delta) = M + L\Delta(I - D\Delta)^{-1}R \quad \text{where } \Delta = \mathbf{diag}(\delta_1 I_{r_1}, \dots, \delta_p I_{r_p}) \quad (31)$$

A LFR is thus a matrix-based way to describe a multivariable rational matrix-valued function. It is a generalization, to the multivariable case, of the well-known state-space representation of transfer functions (i.e. monovariable rational functions).

A constructive proof of the above result is based on a simple idea: first devise LFRs for simple (e.g. linear) functions, then use combination rules (such as multiplication, addition, etc.), to devise LFRs for arbitrary rational functions. To define the combination rules, we will start from two rational matrix-valued functions of  $\delta \in \mathbb{R}^p$ , that are described in the LFR format:

$$\mathbf{M}_i(x) = M_i + L_i \Delta_i (I - D_i \Delta_i)^{-1} R_i, \quad i = 1, 2$$

where  $\Delta_i = \mathbf{diag}(\delta_1 I_{r_1}, \dots, \delta_p I_{r_p})$ ,  $i = 1, 2$ . In the sequel, we define  $\tilde{\Delta} = \mathbf{diag}(\Delta_1, \Delta_2)$ .

*Addition.* The sum of  $\mathbf{M}_1$  and  $\mathbf{M}_2$  has the LFR

$$\mathbf{M}(\delta) = \mathbf{M}_1(\delta) + \mathbf{M}_2(\delta) = M + L\tilde{\Delta}(\delta)(I - D\tilde{\Delta}(\delta))^{-1}R$$

with

$$M = M_1 + M_2, \quad L = [L_1 \quad L_2], \quad R = \begin{bmatrix} R_1 \\ R_2 \end{bmatrix}, \quad D = \mathbf{diag}(D_1, D_2)$$

Using row and column permutations, it is then possible to rewrite  $\mathbf{M}(\delta)$  as in (31) (see the shuffling item below).

*Multiplication.* The product of  $\mathbf{M}_1(\delta)$  and  $\mathbf{M}_2(\delta)$  is given by

$$\mathbf{M}(\delta) = \mathbf{M}_1(\delta)\mathbf{M}_2(\delta) = M + L\tilde{\Delta}(\delta)(I - D\tilde{\Delta}(\delta))^{-1}R$$

where

$$M = M_1M_2, \quad L = [L_1 \quad M_1L_2], \quad R = \begin{bmatrix} R_1M_2 \\ R_2 \end{bmatrix}, \quad D = \begin{bmatrix} D_1 & R_1L_2 \\ 0 & D_2 \end{bmatrix}$$

*Stacking.* The combination of  $\mathbf{M}_1(\delta)$  and  $\mathbf{M}_2(\delta)$  is

$$\mathbf{M}(\delta) = [\mathbf{M}_1(\delta) \quad \mathbf{M}_2(\delta)] = M + L\tilde{\Delta}(\delta)(I - D\tilde{\Delta}(\delta))^{-1}R$$

with

$$M = [M_1 \quad M_2], \quad L = [L_1 \quad L_2], \quad R = \mathbf{diag}(R_1, R_2), \quad D = \mathbf{diag}(D_1, D_2)$$

*Shuffling.* Suppose we are given a matrix function

$$\mathbf{M}(\delta) = M_s + L_s\Delta_s(\delta)(I - D_s\Delta_s(\delta))^{-1}R_s$$

with  $\Delta_s(\delta)$  not necessarily in the required order (the variables may appear shuffled). Then, we can use row and column permutations to put the above representation back into the LFR format. Precisely, take a permutation matrix  $E$  such that for every  $x$ ,

$$E^T\Delta_s(\delta)E = \Delta(\delta) = \mathbf{diag}(x_1I_{r_1}, \dots, x_nI_{r_n}).$$

In this case,  $\mathbf{M}(\delta)$  has the LFR

$$\mathbf{M}(\delta) = M + L\Delta(\delta)(I - D\Delta(\delta))^{-1}R$$

where  $M = M_s$ ,  $L = L_sE$ ,  $R = E^TR_s$  and  $D = E^TD_sE$ .

*Inversion.* If  $\mathbf{M}$  is a square matrix with  $M(0)$  invertible, and has an LFR

$$\mathbf{M}(\delta) = M + L\Delta(\delta)(I - D\Delta(\delta))^{-1}R$$

then its inverse can be written, for every  $\delta$  such that  $\mathbf{M}(\delta)$  is invertible, as

$$\mathbf{M}(\delta)^{-1} = M^{-1} - M^{-1}L\Delta(\delta)(I - (D - RM^{-1}L)\Delta(\delta))^{-1}RM^{-1}$$

## APPENDIX B. LFR OF THE CLOSED-LOOP SYSTEM

The LFR of the closed-loop system is

$$\dot{\tilde{x}} = (\tilde{A} + \tilde{B}K\tilde{C}_y + \tilde{B}_uK_u)\tilde{x} + (\tilde{B}_p + \tilde{B}K\tilde{D}_{yp})\tilde{p} + (\tilde{B}_w + \tilde{B}K\tilde{D}_{yw})w$$

$$\tilde{q} = (\tilde{C}_q + \tilde{D}K\tilde{C}_y + \tilde{D}_{qu}K_u)\tilde{x} + (\tilde{D}_{qp} + \tilde{D}K\tilde{D}_{yp})\tilde{p} + (\tilde{D}_{qw} + \tilde{D}K\tilde{D}_{yw})w$$

$$z = (\tilde{C}_z + \tilde{D}_{zu}K_u)\tilde{x} + \tilde{D}_{zp}\tilde{p}$$

$$\tilde{p} = \tilde{\Delta}\tilde{q}, \quad \tilde{\Delta} = \text{diag}(\Delta^{v \times v}, \Delta_{\text{meas}}^{v_{\text{meas}} \times v_{\text{meas}}})$$

$$\tilde{A} = \begin{bmatrix} A^{n \times n} & 0^{n \times n} \\ 0^{n \times n} & 0^{n \times n} \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} 0^{n \times n} & 0^{n \times n_u} \\ I^{n \times n} & 0^{n \times n_u} \end{bmatrix}, \quad \tilde{B}_u = \begin{bmatrix} B_u^{n \times n_u} \\ 0^{n \times n_u} \end{bmatrix}$$

$$\tilde{B}_p = \begin{bmatrix} B_p^{n \times v} & 0^{n \times v_{\text{meas}}} \\ 0^{n \times v} & 0^{n \times v_{\text{meas}}} \end{bmatrix}, \quad \tilde{C}_y = \begin{bmatrix} 0^{n \times n} & I^{n \times n} \\ 0^{v_{\text{meas}} \times n} & 0^{v_{\text{meas}} \times n} \\ C_y^{n_y \times n} & 0^{n_y \times n} \end{bmatrix}$$

$$\tilde{D}_{qp} = \begin{bmatrix} D_{qp}^{v \times v} & 0^{v \times v_{\text{meas}}} \\ 0^{v_{\text{meas}} \times v} & 0^{v_{\text{meas}} \times v_{\text{meas}}} \end{bmatrix}, \quad \tilde{D}_{yp} = \begin{bmatrix} 0^{n \times v} & 0^{n \times v_{\text{meas}}} \\ 0^{v_{\text{meas}} \times v} & I^{v_{\text{meas}} \times v_{\text{meas}}} \\ D_{yp}^{n_y \times v} & 0^{n_y \times v_{\text{meas}}} \end{bmatrix}$$

$$\tilde{D}_{qu} = \begin{bmatrix} D_{qu}^{v \times n_u} \\ 0^{v_{\text{meas}} \times n_u} \end{bmatrix}, \quad \tilde{C}_q = \begin{bmatrix} C_q^{v \times n} & 0^{v \times n} \\ 0^{v_{\text{meas}} \times n} & 0^{v_{\text{meas}} \times n} \end{bmatrix}, \quad \tilde{D} = \begin{bmatrix} 0^{v \times n} & 0^{v \times v_{\text{meas}}} \\ 0^{v_{\text{meas}} \times n} & I^{v_{\text{meas}} \times v_{\text{meas}}} \end{bmatrix}$$

$$\tilde{B}_w = \begin{bmatrix} B_w^{n \times n_w} \\ 0^{n \times n_w} \end{bmatrix}, \quad \tilde{D}_{yw} = \begin{bmatrix} 0^{n \times n_w} \\ 0^{v_{\text{meas}} \times n_w} \\ D_{yw}^{n_y \times n_w} \end{bmatrix}, \quad \tilde{D}_{qw} = \begin{bmatrix} D_{qw}^{v \times n_w} \\ 0^{v_{\text{meas}} \times n_w} \end{bmatrix}$$

$$\tilde{C}_z = [C_z^{n_z \times n} \quad 0^{n_z \times n}], \quad \tilde{D}_{zu} = D_{zu}^{n_z \times n_u}, \quad \tilde{D}_{zp} = [D_{zp}^{n_z \times n_p} \quad 0^{n_z \times v_{\text{meas}}}]$$

$$K = \begin{bmatrix} \bar{A}^{n \times n} & \bar{B}_p^{n \times v_{\text{meas}}} & \bar{B}_y^{n \times n_y} \\ \bar{C}_q^{v_{\text{meas}} \times n} & \bar{D}_{qp}^{v_{\text{meas}} \times v_{\text{meas}}} & \bar{D}_{qy}^{v_{\text{meas}} \times n_y} \end{bmatrix}, \quad K_u = [0^{n_u \times n} \quad \bar{C}_u^{n_u \times n}]$$