



On the Numerical Solution of Differential Linear Matrix Inequalities

Marco Ariola¹ · Gianmaria De Tommasi² · Adriano Mele³ · Gaetano Tartaglione¹

Received: 16 September 2019 / Accepted: 24 February 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

This paper presents a novel approach for the numerical solution of differential linear matrix inequalities. The solutions are searched in the class of piecewise-quadratic functions with symmetric matrix coefficients to be determined. To limit the numbers of unknowns, congruence constraints are considered to guarantee continuity of the solution and of its derivative. In Example section, some control problems involving differential linear matrix inequalities are considered and solved in order to compare the proposed approach with alternative approximation methods adopted in the literature.

Keywords Differential LMIs · Optimization · Time-varying systems

Mathematics Subject Classification 93C05 · 93B50 · 93D05 · 93C57

Communicated by Firdaus E. Udwardia.

✉ Gaetano Tartaglione
gaetano.tartaglione@uniparthenope.it

Marco Ariola
marco.ariola@uniparthenope.it

Gianmaria De Tommasi
detommas@unina.it

Adriano Mele
adriano.mele@unina.it

¹ Dipartimento di Ingegneria, Università degli Studi di Napoli Parthenope, Centro Direzionale di Napoli, Isola C4, 80143 Naples, Italy

² Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione, Università degli Studi di Napoli Federico II, Via Claudio 21, 80125 Naples, Italy

³ Consorzio di Ricerca per l'Energia, l'Automazione e le Tecnologie dell'Elettromagnetismo, Via Claudio 21, 80125 Naples, Italy

1 Introduction

In this paper, we deal with the problem of numerically solving, and hence approximating, a set of differential linear matrix inequalities (DLMI). DLMI arise in a number of different problems in control theory: in [1,2], DLMI are used to study stability of impulsive linear systems, both deterministic and stochastic; a DLMI approach is adopted to design optimal sampled-data control laws for linear system in [3] and for continuous-time Markov jump linear systems in [4]; in [5], a DLMI approach is introduced for the solution of control design problems under uncertainty, focusing on finite-horizon linear systems. The books [6,7] collect a number of different analysis and design problems, where conditions based on DLMI are used in the context of finite-time stability and stabilization; these conditions are extended to the case of Markov jump linear systems in [8] and to the case of Itô stochastic linear systems in [9]; a possible application of such control techniques can be found in [10].

Typically, DLMI are solved by dividing the time interval in equally spaced subintervals and assuming a piecewise-linear trend in each subinterval; as a consequence, there are some points where the function is not differentiable, and thus additional constraints on the left and right derivatives must be taken into account to guarantee that the DLMI are satisfied everywhere. Moreover, there are some cases when the time derivative of the solution needs to be calculated (see, for instance, chap. 3 in [6]). In this paper, we rather propose to use a piecewise-quadratic (PWQ) trend in each subinterval; indeed, this approach prompts the immediate advantage of allowing one to impose that the solution be differentiable in all the time interval.

The results obtained with this solution will be examined throughout the work. In particular, by means of several examples, we show that the PWQ approximation performs better than the piecewise-linear approximation in terms of: (a) computational time needed to solve a given feasibility problem, (b) minimum cost achieved in optimization problems. The numerical solution of DLMI has been tackled also in [11], where the authors propose methods alternative to piecewise-linear approximation to perform DLMI optimization. In particular, they introduce approximations based on Taylor and Fourier truncated series. Our last example shows that our proposed PWQ approach performs slightly better than those proposed in [11].

In what follows, first we formally introduce the family of PWQ solutions that we will consider in the remainder of the paper. The class of PWQ solutions is then used in Sect. 3 to numerically solve the optimization or feasibility problems with DLMI constraints involved in various control problems. It will be shown that the use of PWQ class improves the *quality* of the solution, i.e., it permits to better approximate the optimal solution of the considered problem and, at the same time, it permits to reduce both the computational burden and the memory required to solve the problem, when compared with other possible approaches. Moreover, in the optimization problems, fixing the computational burden, the PWQ approximation permits to achieve better results. Finally, in Sect. 4, some conclusive remarks are drawn.

2 Numerical Solution of DLMI

In this section, we introduce the class of piecewise-quadratic solutions that we will use in the next section to solve optimization problems that involve DLMI constraints.

Let us consider the following optimization problem over the finite time interval $\Delta = [t_0, t_f]$

$$\min_{X(\cdot)} \mathcal{F}(X(t)), \quad t \in \Delta, \tag{1}$$

subject to

$$\mathcal{G}(\dot{X}(t), X(t)) < 0, \quad t \in \Delta, \tag{2a}$$

$$\mathcal{H}(X(t)) < 0, \quad t \in \Delta, \tag{2b}$$

where $X(\cdot) : \Delta \mapsto \mathbb{R}^{v \times v}$ is a continuously differentiable matrix-valued function, while $\mathcal{F}(\cdot)$ is the cost function to be minimized. The constraints $\mathcal{G}(\cdot, \cdot)$ in (2) are the DLMI ones, i.e., a set of linear matrix inequalities that involve both $X(\cdot)$ and its time derivative $\dot{X}(\cdot)$, while $\mathcal{H}(\cdot)$ contains the possibly remaining LMI constraints on $X(\cdot)$.

The DLMI constraints (2a) need to be rewritten as LMIs in order to numerically solve problem (1)-(2), by means of *off-the-shelf* optimization software tools, such as SeDuMi [12], SDPT3 [13] or MOSEK [14]. Different approaches can be adopted to recast DLMI into LMIs, by looking to specific class of solutions to problems (1)-(2). The most common and straightforward approach is to assume the matrix-valued function $X(\cdot)$ to be piecewise affine (PWA) (among the various authors that proposed such an approach see [4–6, 11]). If this is the case, then the time interval Δ is divided into $n_s = \Delta/T_s$ subintervals, where T_s represents the chosen sampling time, and in each subinterval, $X(\cdot)$ is assumed equal to

$$X(t) = \begin{cases} X_i + \Theta_i(t - iT_s - t_0), & \text{for } t \in [t_0 + iT_s, t_0 + (i + 1)T_s), \quad i = 0, \dots, n_s - 2, \\ X_{n_s-1} + \Theta_{n_s-1}(t - (n_s - 1)T_s - t_0), & \text{for } t \in [t_0 + (n_s - 1)T_s, t_f], \end{cases}$$

being $X_i, \Theta_i \in \mathbb{R}^{v \times v}$ the optimization variables, and the following *congruence* constraints must be added to guarantee continuity of $X(\cdot)$ in Δ

$$X_i + T_s \Theta_i = X_{i+1}, \quad i = 0, \dots, n_s - 2.$$

Hence, given n_s subintervals, the PWA solution implies the definition of $n_s + 1$ optimization matrices in $\mathbb{R}^{v \times v}$. However, the PWA class is not included within the continuously differentiable solutions to (1)-(2). It follows that the optimization problem should be *overconstrained* when looking for a PWA solution, therefore introducing further conservatism. Indeed, the DLMI constraints must be fulfilled for both the left

and right derivatives of $X(\cdot)$ at the extrema of each subinterval, i.e., inequalities (2a) must be replaced by the following additional LMIs

$$\mathcal{G}(\Theta_i, X_i) < 0, \quad (3a)$$

$$\mathcal{G}(\Theta_i, X_i + T_s \Theta_i) < 0, \quad (3b)$$

for each $i = 0, \dots, n_s - 1$.

In order to overcome these limitations, in this paper we propose to solve problems (1)–(2) by searching for PWQ solutions. Hence, in each subinterval, the unknown matrix-valued function $X(\cdot)$ is assumed to be equal to

$$X(t) = \begin{cases} X_i + \Theta_i (t - iT_s - t_0) + \Xi_i (t - iT_s - t_0)^2, \\ \quad \text{for } t \in [t_0 + iT_s, t_0 + (i+1)T_s), \quad i = 0, \dots, n_s - 2, \\ X_{n_s-1} + \Theta_{n_s-1} (t - (n_s - 1)T_s - t_0) + \Xi_{n_s-1} (t - (n_s - 1)T_s - t_0)^2, \\ \quad \text{for } t \in [t_0 + (n_s - 1)T_s, t_f], \end{cases}$$

where $X_i, \Theta_i, \Xi_i \in \mathbb{R}^{\nu \times \nu}$ are the optimization variables. Similarly to what has been done for the PWA solution, the following congruence constraints must be added to ensure continuity of the solution

$$X_i + T_s \Theta_i + T_s^2 \Xi_i = X_{i+1}, \quad i = 0, \dots, n_s - 2.$$

Moreover, the PWQ class permits also to guarantee continuity of the derivative, by means of the following additional constraints

$$\Theta_i + 2T_s \Xi_i = \Theta_{i+1}, \quad i = 0, \dots, n_s - 2. \quad (4)$$

Therefore, when dealing with PWQ solutions, there is no need to *duplicate* the DLMI constraints (2a) as in the PWA case. It should be noticed that, differently from (3), being congruence constraints, Equations (4) do not introduce additional conservatism. Furthermore, given the congruence constraints, the number of $\nu \times \nu$ optimization matrices needed to describe a PWQ solution in n_s subintervals is equal to $n_s + 2$, just one more than the ones needed for a PWA solution.

Despite the additional conservatism attached to the PWA class, both the PWA and PWQ solutions can be used to approximate a general continuous matrix function $X(\cdot)$ with adequate accuracy, provided that T_s is sufficiently small. Since the use of the latter class rather than the former generally guarantees a better accuracy by considering a similar or lower number of optimization matrices, it may result in a decrease in the computational burden, as it will be shown in the next section.

3 Application to Control Problems

In this section, we consider different control problems involving DLMI feasibility conditions, in order to investigate the computational efficiency and performance of

the proposed DLMI solving approach. As discussed in Sect. 2, the proposed approach is based on a PWQ approximation of the solution, and it will be compared with alternative approximation techniques, such as a PWA approximation or the ones based on *truncated Taylor or Fourier series* considered in [11].

The problems considered in this section are the following:

1. the finite-time stability (FTS) conditions of a linear time-varying (LTV) system proposed in [6];
2. the finite-horizon state-feedback control problem appeared in [5];
3. the \mathcal{H}_∞ sampled-data state-feedback control problem appeared in [4].

All the results presented in this section have been obtained by recasting the DLMIs in the original problem as LMIs, and by solving them using MOSEK [14], a commercial optimization software that deals with semidefinite programming and has a MATLAB API accessible via the YALMIP parser [15]. The setup of the machine that has been used is as follows: 2.5 GHz Intel Core i5-7200U processor and 8GB 1600 MHz DDR3 running the MATLAB R2017b version.

Example 3.1 (FTS assessment for LTV systems) The aim of this example is to compare the computational burden related to the PWA and to the PWQ approximations when solving DLMIs.

Taking into account the FTS conditions appeared in [6, Theorem 2.1], the LTV system

$$\dot{x}(t) = A(t)x(t), \quad x(t_0) = x_0, \tag{5}$$

is finite-time stable with respect to $(t_0, T, R, \Gamma(\cdot))$ iff there exists a symmetric matrix function $P(\cdot)$, which satisfies, for all $t \in [t_0, t_0 + T]$, the following DLMIs/LMIs feasibility problem

$$\dot{P}(t) + A^T(t)P(t) + P(t)A(t) < 0, \tag{6a}$$

$$P(t) > \Gamma(t), \tag{6b}$$

$$P(t_0) < R. \tag{6c}$$

In this example, we have considered several random systems of order $n = 2, \dots, 5$. For each system, the dynamic matrix was chosen as $A(t) = A_0 + atI_n$, where A_0 and a are a randomly generated $n \times n$ matrix and a random scalar, respectively, being I_n the $n \times n$ identity matrix. The sampling time T_s has been assumed to belong to the following set

$$T_s \in \left\{ \frac{1}{32}, \frac{1}{16}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}, 1 \right\}. \tag{7}$$

For each n we have considered 100 systems which admitted a *piecewise-quadratic* solution to the feasibility problem (6), with the following FTS parameters

Table 1 Example 3.1: number of PWA failures solving 100 DLMI problems (6) for each order n by using MOSEK. Note that all the considered DLMI problems admit a solution when using the PWQ family of solutions

n	2	3	4	5
# PWA failures	28	45	53	83

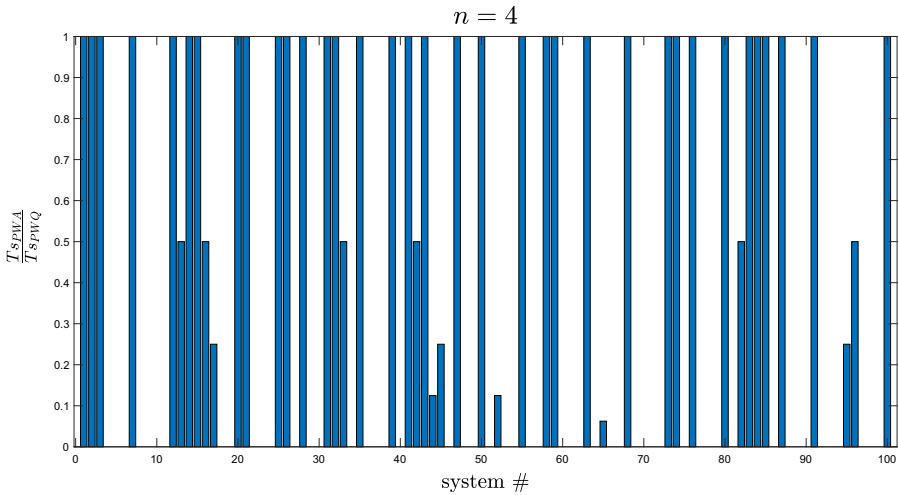


Fig. 1 Example 3.1: comparison between the sampling times needed to solve the DLMI problem (6) using a *piecewise-affine* approximation and a *piecewise-quadratic* approximation. A null value indicates a problem that cannot be solved with the piecewise-affine approximation

$$t_0 = 0, \quad T = 2, \quad R = I_n, \quad \Gamma(t) = 0.1e^{-t/3} \cdot I_n,$$

and for a value of T_s belonging to set (7).

The PWQ solution to the considered FTS problem was evaluated by discretizing the DLMI with the approach described in Sect. 2 and by using the MOSEK solver.

The same DLMI feasibility problem was then solved looking for a PWA solution. Table 1 shows the number of failures of the PWA approach in solving the DLMI problem (6). A PWA failure indicates that the problem could not be solved with the PWA approach, even using the minimum sampling time in set (7). It should indeed be considered that the advantage given by the higher number of *degrees of freedom* in the PWA case for smaller sampling times is partially *neutralized* by the increased number of constraints that need to be added to fulfill conditions of type (2a) by both left and right derivatives (see also the discussion in Sect. 2). The results also show that the number of failures of the PWA approximation increases with the order of the LTV systems.

Next, we consider a comparison between the computational burden needed to solve the DLMI problem (6) using the MOSEK solver for computing a PWA and a PWQ approximation of the solution, respectively. In particular, we analyze the sampling times and the computational times needed to compute the two solutions.

Table 2 Example 3.1: smaller sampling time for solving DLMI in Example 3.1. The table considers only the cases when both approaches give a solution; in the other cases, the PWA does not provide a solution

n	2	3	4	5
PWQ	0	0	0	0
PWA	8	6	12	4
same	64	49	35	13

Table 3 Smaller computational time for solving DLMI in Example 3.1. The table considers only the cases when both approaches give a solution; in the other cases, the PWA does not provide a solution

n	2	3	4	5
PWQ	55	48	44	17
PWA	17	7	3	0

Figure 1 shows the ratio between the sampling times T_s used to discretize and solve the 100 DLMI problems (6) for $n = 4$ considering either a PWA or a PWQ solution. The comparison for each order n is synthesized in Table 2, which indicates how many times each approach has required a smaller sampling time, for the cases in which a solution has been found by means of the PWA approach. It looks evident that the sampling time needed to find the PWA solutions is never higher than the one needed in the PWQ case, which implies that the number of LMIs needed to compute a PWA solution is always higher. Furthermore, also the number of unknowns needed to compute a PWA solution is higher when the sampling time is smaller than the one needed in the PWQ case.

Based on the above results, we analyze the computational time needed to solve the DLMI by using the two approaches. Figure 2 shows the ratio between the computational times for the considered LTV systems for $n = 4$. A null value indicates a problems that cannot be solved with the piecewise-affine approximation also using the chosen minimum sampling time. The comparison for each order n is synthesized in Table 3, which indicates how many times one approach needed a smaller computational time than the other, for the cases in which a solution has been found by means of the PWA approach. It can be noted that, even when the PWA approach did not fail, it generally needed a larger computational time to find the solution to the considered FTS problem.

Example 3.2 (Estimation of the FTS initial domain) The aim of this example is to compare the conservativeness implied by the class of PWA and PWQ solutions used to solve a problem that involves DLMI constraints.

We focus again on the FTS problem (6) applied to the LTV system (5), which has been introduced in the previous example. This time, the following parameters are assumed for the DLMI feasibility problem

$$t_0 = 0, \quad T = 2, \quad R = \beta \cdot I, \quad \Gamma(t) = 0.25e^{-t/3} \cdot I.$$

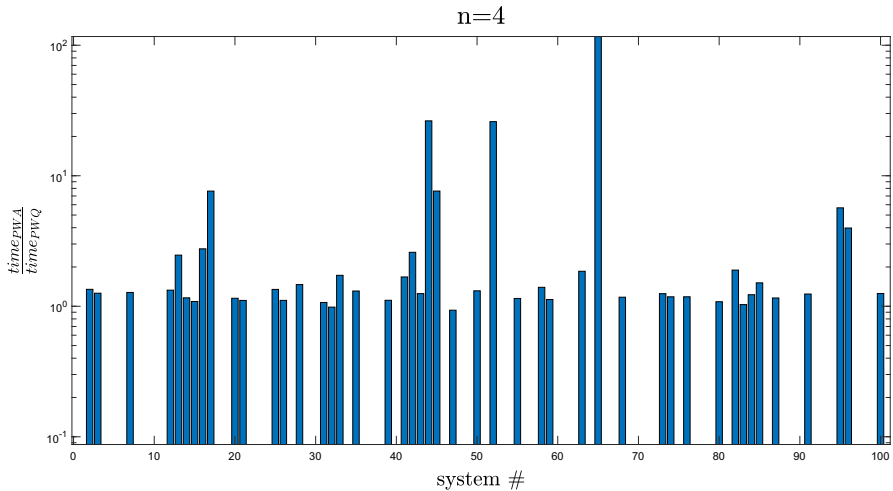


Fig. 2 Example 3.1: comparison on a logarithmic axis between the computational times needed to solve the DLMI problem (6) using a *piecewise-affine* approximation and a *piecewise-quadratic* approximation. A null value indicates a problem that cannot be solved with the piecewise-affine approximation

Letting $R = \beta \cdot I$, we can estimate, by means of a linear search, the minimum value β_{\min} in $[0.25, 5]$, such that the DLMI (6) can be solved; $1/\beta_{\min}$ corresponds to the maximum radius of the circular initial domain such that the system is FTS with respect to $(t_0, T, R, \Gamma(t))$.

A set of random LTV systems (5) of order $n = 2, 3, 4$ were considered, defined as in Example 3.1. In particular, we considered 300 systems for which a PWA solution with sampling time $T_s = 0.1$ and $0.25 \leq \beta \leq 5$ could be found. For these systems, the same DLMI feasibility problem has been solved, searching for a PWQ solution with $0.25 \leq \beta \leq 5$. The resulting values of β_{\min} have been finally compared. The outcome of this analysis, carried out by using the MOSEK solver, can be summarized as follows:

1. The PWA approximation never provided a larger maximum radius estimate;
2. In the 46% of the cases, the PWQ approximation provided a larger maximum radius estimate;
3. In the 54% of the cases, the two approaches obtained a same result.

Example 3.3 (Optimal control design) Another comparison between the use of a PWA and the proposed PWQ solution was carried out by considering the finite-horizon, time-varying, state-feedback design problem appeared in [5], which involves the DLMI/LMI feasibility problem described below.

Let us consider the LTV system

$$\dot{x}(t) = A(t)x(t) + B_1(t)w(t) + B_2(t)u(t), \quad x(0) = 0 \tag{8a}$$

$$z(t) = C_1(t)x(t) + D_{12}(t)u(t), \tag{8b}$$

$$u(t) = K(t)x(t), \tag{8c}$$

where $x(\cdot) \in \mathbb{R}^n$ is the state vector, $u(\cdot) \in \mathbb{R}^l$ represents the control action and $z(\cdot) \in \mathbb{R}^m$ is the state combination to be attenuated. The exogenous disturbance signal $w(\cdot) \in \mathbb{L}_2^q[0, T]$ is supposed to be a vector signal of dimension q , square-summable in the time interval $[0, T]$. The matrices $A(\cdot), B_1(\cdot), B_2(\cdot), C_1(\cdot), D_{12}(\cdot)$ are time-varying matrices of appropriate dimensions.

The problem consists in finding the state-feedback gain matrix $K(\cdot)$ which, for a prescribed scalar $\gamma > 0$, for a given $P_T > 0$ and for all $w(\cdot) \in \mathbb{L}_2^q[0, T]$ minimizes the performance index

$$J(w, 0, T) = x^T(T)P_Tx(T) + \int_0^T (z^Tz - \gamma^2w^Tw) d\tau \leq 0. \tag{9}$$

In [5], the authors show that the above control design problem is equivalent to the following optimization problem, where time dependence is omitted for the sake of brevity

$$\min_{P(\cdot), K(\cdot)} \text{trace}(P(t)), \quad t \in [0, T] \tag{10a}$$

subject to:

$$\begin{bmatrix} \Delta_P & PB_1 & (C_1 + KD_{12})^T \\ * & -\gamma^2I_q & 0 \\ * & * & -I_m \end{bmatrix} \leq 0, \tag{10b}$$

$$\text{where: } \Delta_P = \dot{P} + (A + B_2K)^T P + P(A + B_2K)$$

$$P(t) \geq 0, \tag{10c}$$

$$P(T) = P_T, \tag{10d}$$

from which a sequence of convex programming problems can be obtained by considering a PWA solution.

In particular, once the sampling time T_s has been fixed and the subsequent number of subintervals $N = T/T_s$ has been computed, constraint (10b) can be discretized by considering a PWA approximation of the solution, and the discretized state-feedback problem becomes a linear one consisting in finding the sequence (P_{k-1}, K_k) such that, for $k = N, N - 1, \dots, 1$ and a given P_k , the obtained LMI with respect to P_{k-1} and K_k is satisfied with P_{k-1} of minimum trace.

Here, we propose to find a PWQ solution to the optimization problem (10). In order to linearize constraint (10b), we introduce the matrix-valued optimization variables $X(\cdot) = P^{-1}(\cdot)$ and $Y(\cdot) = K(\cdot)X(\cdot)$, and then we pre- and post-multiply the inequality (10b) by

$$\begin{bmatrix} P^{-1}(\cdot) & 0 & 0 \\ 0 & I_q & 0 \\ 0 & 0 & I_m \end{bmatrix},$$

obtaining the following convex programming problem (time dependence is again omitted)

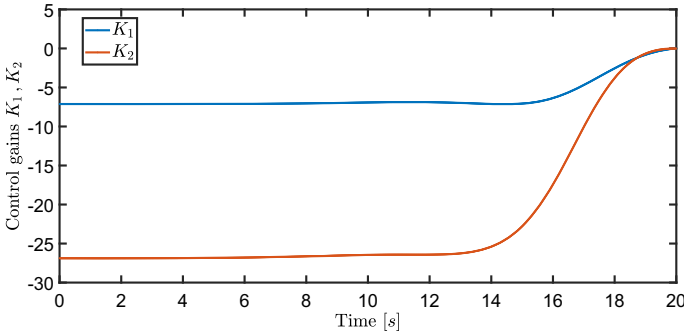


Fig. 3 Example 3.3: the evolution in time of $K_{PWQ}(t) = [K_1(t) \ K_2(t)]$ for $t \in [0, T]$, by considering the parameters (12). These gains have the same time behavior of those shown in [5, Fig. 1], but they have been obtained using a double sampling time T_s

$$\max_{X(\cdot), Y(\cdot)} \text{trace}(X(t)), \quad t \in [0, T] \tag{11a}$$

subject to:

$$\begin{bmatrix} \Delta_X & B_1 & (C_1 X + K D_{12} Y)^T \\ * & -\gamma^2 I_q & 0 \\ * & * & -I_m \end{bmatrix} \leq 0, \tag{11b}$$

where: $\Delta_X = -\dot{X} + XA^T + AX + Y^T B_2^T + B_2 Y$

$$X(t) \geq 0, \tag{11c}$$

$$X(T) = P_T^{-1}, \tag{11d}$$

With this transformation, the same procedure used to find the sequence (P_{k-1}, K_k) can be applied to find a solution in terms of (X_{k-1}, Y_k) , with the only difference that the trace of X_k must be maximized. The optimal state-feedback control gain matrix is then given by $K(t) = Y(t)X^{-1}(t)$.

To compare the two approaches, we consider the control design problem [5, Example 1] characterized by the parameters

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_1 = B_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{12a}$$

$$C_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad D_{12} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{12b}$$

$$T = 20 \text{ s}, \quad \gamma = 1.01. \tag{12c}$$

In order to find a PWA solution to the control design problem, in [5] the sampling time was fixed to $T_s = 0.005 \text{ s}$, resulting in a number of subintervals of $N = 4000$.

By repeating the same analysis with a PWQ approach, it was found that it is possible to solve the same problem by using the MOSEK solver and by considering a sampling time $T_s = 0.01 \text{ s}$ and a number of subintervals $N = 2000$. The evolution in time of $K_{PWQ}(t) = [K_1(t) \ K_2(t)]$ for $t \in [0, T]$ is shown in Fig. 3. To compare the PWQ

solution with the PWA one, we first consider the control gains at the initial time. By using the PWQ approach, we computed the gains $K_{PWQ}(0) = [-7.1233 \ -26.8888]$, whereas in [5, Example 1] the values $K_{PWA}(0) = [-7.1235 \ -26.8897]$ were obtained by using the PWA approach.

Moreover, a comparison with the gains shown in [5, Fig. 1] immediately proves that the PWQ approach allows us to compute an equivalent solution, but with a larger sampling time and hence a smaller computational time.

Example 3.4 (Minimization of \mathcal{H}_∞ norm) The aim of this last example is to compare the PWQ solution to the DLMI problem with the alternative approaches proposed in [11]. In particular, the authors in [11] propose to define the optimal solution of a DLMI feasibility problem as a Taylor polynomial expansion of finite degree or as a truncated Fourier series. It is worth to notice that the approaches considered in [11] rely on Taylor and Fourier expansions on the whole time interval, while the PWQ approach requires to divide the time interval in subintervals.

In order to investigate the proposed methods as well as to evaluate and compare their numerical efficiency and limitations, in [11] the authors consider a series of optimal sampled-data control design problems of increasing complexity. The one considered here regards the minimization of the \mathcal{H}_∞ norm from the exogenous input w to the controlled output z of a closed-loop linear system governed by a state-feedback sampled-data control law given by

$$\dot{x}(t) = Ax(t) + Bu(t) + Ew(t) \tag{13a}$$

$$z(t) = Cx(t) + Du(t) \tag{13b}$$

$$u(t) = Lx(t_k) \quad \forall t \in [t_k, t_{k+1}), \tag{13c}$$

where $t_0 = 0, t_{k+1} - t_k = h$ are evenly spaced sampling instants for all $k \in \mathbb{N}$ with sampling period $h > 0$, and $x(\cdot) \in \mathbb{R}^n, u(\cdot) \in \mathbb{R}, w(\cdot) \in \mathbb{R}$ and $z(\cdot) \in \mathbb{R}^{n+1}$ are the state, the control, the exogenous input and the controlled output, respectively. The sampled-data state-feedback gain matrix $L \in \mathbb{R}^{1 \times n}$ is the design variable to be determined.

In [11], the authors show that the problem of finding the optimal \mathcal{H}_∞ sampled-data state-feedback control problem is equivalent to the following convex programming problem

$$\min_{X(\cdot), Y, V, \mu} \mu, \tag{14a}$$

subject to:

$$\begin{bmatrix} -\dot{X}(t) + X(t)F^T + FX(t) & X(t)G^T & J \\ * & -I & 0 \\ * & * & -\mu I \end{bmatrix} < 0, \tag{14b}$$

$$\begin{bmatrix} V & [V \ Y^T] \\ * & X(t_0) \end{bmatrix} > 0, \tag{14c}$$

$$\begin{bmatrix} X(h) & X(h) \begin{bmatrix} I \\ 0 \end{bmatrix} \\ * & V \begin{bmatrix} I \\ 0 \end{bmatrix} \end{bmatrix} > 0, \quad (14d)$$

where

$$F = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, \quad G = [C \ D], \quad J = \begin{bmatrix} E \\ 0 \end{bmatrix},$$

μ is a scalar variable, Y and V are a full and a symmetric matrix variable, respectively, while $X(\cdot)$ is a symmetric matrix-valued function. If a solution to the convex optimization problem (14) exists, then the optimal state feedback control gain matrix is given by $L = YV^{-1}$, and μ is the minimum \mathcal{H}_∞ cost.

For the sake of completeness, it should be noted that the sampled-data control framework presented so far has been further extended to the output feedback case in [3]. However, in this example, we refer to the state-feedback case, which allows us to compare our approach with results in [11].

Let us consider the open-loop unstable system proposed in [11] and given by

$$A = \begin{bmatrix} 0_{n-1} & I_{n-1} \\ 0 & 0_{n-1}^T \end{bmatrix}, \quad B = \begin{bmatrix} 0_{n-1} \\ 1 \end{bmatrix}, \quad (15a)$$

$$C = \begin{bmatrix} I_n \\ 0_n^T \end{bmatrix}, \quad D = \begin{bmatrix} 0_n \\ 1 \end{bmatrix}, \quad E = 1_n \quad (15b)$$

where I_n , 0_n and 1_n denote the identity $n \times n$ matrix, the null $n \times 1$ vector and the one $n \times 1$ vector, respectively. Then, we have solved the convex problem (13) by setting $h = 1$ s.

To compare the performance of the different approaches, we solved the convex programming problem for an increasing system order $n = 2, \dots, 5$, computing the PWQ optimal solutions for different numbers of subintervals N by using the MOSEK solver. The number of subintervals was chosen in order to compare the results with those presented in [11, Section IV], where the computational burden was analyzed in terms of the number of evaluations m of the derivative $\dot{X}(\cdot)$. In particular, in [11] it is shown that in the PWA case $m = 2N$, while in the case of a solution defined as a truncated series $m \approx 3n_\phi$, where n_ϕ is the number of terms in the series, excluding the constant one. Now, by considering the discretization method presented in Sect. 2, we have that in the PWQ case $m = N + 1$. In this way, the comparison between the PWQ solution and the ones presented in [11, Table II and Table III] can be carried out at the same computational burden, in terms of m , by evaluating the minimum \mathcal{H}_∞ cost of Problem (14).

Tables 4 and 5 summarize the minimum \mathcal{H}_∞ cost of Problem (14) by considering the different discretization approaches and different pairs of values (m, n) . It can be noticed that at the same computational burden (i.e., once m is fixed), the PWQ solution allowed us to compute a lower value for the \mathcal{H}_∞ cost in all cases.

Table 4 Example 3.4: minimum \mathcal{H}_∞ cost of problem (14) for PWQ and PWA solutions. The costs for the PWA solution are those presented in [11, Table I]

m/n		2	3	4	5
2	PWQ	52.36	632.23	1.11×10^4	5.65×10^4
	PWA	527.35	2.82×10^4	∞	∞
4	PWQ	19.77	116.87	667.98	4.33×10^3
	PWA	122.96	2.70×10^3	8.53×10^4	∞
8	PWQ	15.47	78.65	357.63	1.58×10^3
	PWA	46.01	515.46	7.14×10^3	1.42×10^5
16	PWQ	14.23	69.85	299.75	1.22×10^3
	PWA	25.90	187.95	1.45×10^3	1.40×10^4
32	PWQ	13.76	67.03	283.12	1.13×10^3
	PWA	19.12	111.41	617.47	3.68×10^3

Table 5 Example 3.4: minimum \mathcal{H}_∞ cost of problem (14)–(15) for a PWQ solution and solutions defined as truncated Taylor and Fourier series. The costs for truncated Taylor and Fourier series cases are those presented in [11, Table II and Table III]

m/n		2	3	4	5
3	PWQ	25.04	175.65	1.32×10^3	1.31×10^4
	TAYLOR	527.35	2.82×10^4	∞	∞
	FOURIER	1.24×10^3	1.09×10^5	∞	∞
6	PWQ	16.58	87.52	421.61	2.04×10^3
	TAYLOR	55.21	730.24	1.35×10^4	3.72×10^5
	FOURIER	101.69	1.92×10^3	5.61×10^4	∞
9	PWQ	15.16	76.28	341.40	1.47×10^3
	TAYLOR	22.14	137.33	863.79	6.62×10^3
	FOURIER	32.59	258.52	2.42×10^3	3.25×10^4
12	PWQ	14.60	72.28	315.01	1.31×10^3
	TAYLOR	17.63	92.68	436.963	2.07×10^3
	FOURIER	21.61	129.02	744.17	4.86×10^3
15	PWQ	14.30	70.30	302.52	1.24×10^3
	TAYLOR	15.71	78.28	340.71	1.41×10^3
	FOURIER	17.58	92.33	436.18	2.03×10^3

4 Conclusions

A novel approach for the numerical solution of DLMI-based problems has been presented, where PWQ solutions with symmetric matrix coefficients to be determined have been considered. The PWQ class of solutions allows us to take into account congruence constraints to guarantee the continuity of both the solution and its derivative.

In this way, (i) we limit the numbers of unknowns; (ii) we obtain a well-posed problem, when dealing with control design approaches that define the controller matrices as functions of the derivative of the solution [16]. Some control problems, involving differential linear matrix inequalities, have been considered and solved in order to compare the proposed approach with alternative approximation methods adopted in the literature. Numerical results have shown that the PWQ approximation performs better in terms of both the computational time needed to solve a given feasibility problem and the minimum cost achieved in optimization problems.

References

1. Briat, C.: Convex conditions for robust stability analysis and stabilization of linear aperiodic impulsive and sampled-data systems under dwell-time constraints. *Automatica* **49**(11), 3449–3457 (2013)
2. Briat, C.: Stability analysis and stabilization of stochastic linear impulsive, switched and sampled-data systems under dwell-time constraints. *Automatica* **74**, 279–287 (2016)
3. Geromel, J., Colaneri, P., Bolzern, P.: Differential linear matrix inequality in optimal sampled-data control. *Automatica* **100**, 289–298 (2019)
4. Gabriel, G., Gonçalves, T., Geromel, J.: Optimal and robust sampled-data control of Markov jump linear systems: a differential LMI approach. *IEEE Trans. Autom. Control* **63**(9), 3054–3060 (2018)
5. Shaked, U., Suplin, V.: A new bounded real lemma representation for the continuous-time case. *IEEE Trans. Autom. Control* **46**(9), 1420–1426 (2001)
6. Amato, F., Ambrosino, R., Ariola, M., Cosentino, C., De Tommasi, G.: *Finite-Time Stability and Control*. Springer, Berlin (2014)
7. Amato, F., De Tommasi, G., Pironti, A.: *Finite-Time Stability: An Input–Output Approach*. Wiley, New York (2018)
8. Tartaglione, G., Ariola, M., Amato, F.: An observer-based output feedback controller for the finite-time stabilization of Markov jump linear systems. *IEEE Control Syst. Lett.* **3**(3), 763–768 (2019)
9. Tartaglione, G., Ariola, M., De Tommasi, G., Amato, F.: Annular finite-time stabilization of stochastic linear time-varying systems. In: *Proceedings of the IEEE Conference on Decision and Control, Miami Beach (FL), USA*, pp. 7219–7224 (2018)
10. Ariola, M., De Tommasi, G., Tartaglione, G., Amato, F.: Hybrid architecture for vehicle lateral collision avoidance. *IET Control Theory Appl.* **12**(14), 1941–1950 (2018)
11. Gonçalves, T., Gabriel, G., Geromel, J.: Differential linear matrix inequalities optimization. *IEEE Control Syst. Lett.* **3**(2), 380–385 (2019)
12. Labit, Y., Peaucelle, D., Henrion, D.: SEDUMI INTERFACE 1.02: a tool for solving LMI problems with SEDUMI. In: *Proceedings of IEEE International Symposium on Computer Aided Control System Design*, pp. 272–277 (2002)
13. Toh, K.C., Todd, M.J., Tütüncü, R.H.: SDPT3: a MATLAB software package for semidefinite programming, version 1.3. *Optim. Methods Softw.* **11**(1–4), 545–581 (1999)
14. Andersen, E., Andersen, K.: The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In: *High Performance Optimization*. Springer, pp. 197–232 (2000)
15. Löfberg, J.: YALMIP: a toolbox for modeling and optimization in MATLAB. In: *In Proceedings of the CACSD Conference*. Taipei, Taiwan (2004)
16. Amato, F., Carannante, G., De Tommasi, G., Pironti, A.: Input-output finite-time stability of linear systems: necessary and sufficient conditions. *IEEE Trans. Autom. Control* **57**(12), 3051–3063 (2012)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.