Joint 48th IEEE Conference on Decision and Control and
28th Chinese Control Conference
Shanghai, P.R. China, December 16-18, 2009

Linear Control of Time-Domain Constrained Systems

W.H.T.M. Aangenent, W.P.M.H. Heemels, M.J.G. van de Molengraft, and M. Steinbuch

Abstract— **Recent results on the control of linear systems subject to time-domain constraints could only handle the case of closed-loop poles that are situated on the real axis. As most closed-loop systems in practice contain also complex poles, there is a strong need for a general framework encompassing all cases. In this paper such a framework is presented based on sums-of-squares techniques and we show indeed that time-domain constraints on closed-loop signals of linear systems can be incorporated as linear matrix inequalities, even when complex conjugate poles are assigned. The effectiveness of this complete design method is evaluated by means of a simulation example.**

I. Introduction

The transient response to reference commands or disturbance inputs is an important performance qualifier in many control systems. However, most control design strategies cannot cope directly with requirements on time-domain signals such as tracking errors and control inputs, especially in the continuous-time case. A commonly used method to capture the essence of time-domain specifications is the reformulation into frequency domain requirements [1]. In general, such reformulations are either approximate, conservative or both. As an alternative approach, some specific types of constraints, such as saturation of the input signal, can be dealt with after the control design, for instance by means of anti-windup schemes [2] that results in an overall nonlinear controller. However, it is clear that these techniques cannot handle the general problem in which also state and output signals are constrained and which do not result in linear controllers.

An elegant methodology to enforce time-domain constraints on the input and output of a continuous-time linear control system is presented recently in [3], where linear matrix inequality (LMI) techniques are used to synthesize a fixed order *linear* controller that satisfies the constraints. In [3], the assignment of closed-loop poles is performed by assigning a characteristic polynomial. This polynomial can be easily computed from fractional representations of the transfer functions of the controller and the plant, and allows for a parametrization of all stabilizing controllers using the Youla-Kučera parametrization [4]. After a controller has been designed that achieves the prescribed closed-loop pole locations, the degrees of freedom of the Youla-Kučera parametrization are used to enforce certain time-domain constraints, such as bounds on the input amplitude and output overshoot, exploiting sums-of-squares techniques [5]–[8]. Unfortunately, the approach in [3] is limited to the assignment of distinct strictly negative *real* closed-loop poles, which is a severe restriction in the case of many practical situations such as for lightly damped systems.

The above clearly indicates the need for a general framework encompassing arbitrary closed-loop pole placement, as will be developed in this paper. In particular, we propose an extension to the method in [3], which leads to a general design framework based on sums-of-squares techniques and we show indeed that the resulting linear controller satisfies the time-domain constraints on closed-loop signals, even when complex conjugate poles are assigned. This framework is based on a relaxation, which can solve the constrained control problem at hand with arbitrary accuracy and with full guarantees on the constraints of the system. The conditions can be formulated as linear matrix inequalities. In addition to constraint satisfaction, we will also include an objective function in the convex programming problem that can for instance be used to minimize steady state (tracking) errors and reducing overshoot. As a consequence, the ideas presented in this paper lead to a complete design framework that offers guarantees on constraint satisfaction. Its efficiency will be demonstrated by means of a simulation example.

II. Methodology involving real poles

In [3] a method is presented to incorporate time-domain constraints on input and output signals of a linear system. In this section, we shortly review this procedure for completeness and self-containedness.

Consider the control system depicted in Fig. 1 with a linear strictly proper single-input-single-output plant P given in transfer function notation by $P(s) = \frac{b(s)}{a(s)}$ where $a(s)$ and $b(s)$ are polynomials in the Laplace variable s. The controller C, which is to be designed, is described accordingly by $C(s) = \frac{d(s)}{c(s)}$ resulting in the complementary sensitivity given by

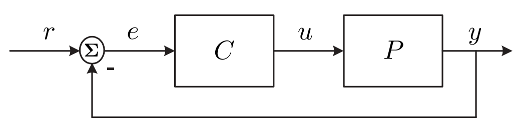

Fig. 1. Block diagram of the closed-loop system with controller C, plant P, and reference signal r, control output signal u, and output signal y.

$$T(s) = \frac{y(s)}{r(s)} = \frac{b(s)d(s)}{a(s)c(s) + b(s)d(s)}. \tag{1}$$

If $a(s)$ and $b(s)$ are coprime (i.e., their greatest common divisor is 1), then arbitrary pole placement can be achieved by designing the corresponding controller polynomials. This is done by solving the polynomial Diophantine equation

$$a(s)c(s) + b(s)d(s) = z(s), \tag{2}$$

where $z(s) = (s+p_1)(s+p_2)\dots(s+p_n)$ is the polynomial with given roots $-p_1,\dots,-p_n$, which are the desired poles of the closed-loop system. There are infinitely many solutions to (2), but there is a unique solution pair $(c_0(s), d_0(s))$ such that $\deg d_0(s) < \deg a(s)$. In this case we have that $d_0(s)$ is of minimal order and as such, $(c_0(s), d_0(s))$ is called the d-minimal solution pair. All possible solutions to the Diophantine equation can then be written as

$$\begin{aligned} c(s) &= c_0(s) + b(s)q(s), \\ d(s) &= d_0(s) - a(s)q(s), \end{aligned} \tag{3}$$

where $q(s)$ is an arbitrary polynomial such that $c_0(s) + b(s)q(s)$ is non-zero. This polynomial, called the Youla-Kučera parameter [9], creates extra freedom in the design of the controller. While the closed-loop poles are invariant for any choice of the Youla-Kučera parameter, the Youla-Kučera parameter enables placement of closed-loop zeros to alter the response. Only proper controllers are considered and therefore there is a degree constraint on $q(s)$. Since the plant was assumed to be strictly proper, and under the additional assumption that $\deg z(s) \geq 2\deg a(s) - 1$ (to enable arbitrary pole placement), this constraint is given as in [10] by

$$\deg q(s) \leq \deg z(s) - 2\deg a(s). \tag{4}$$

The extra freedom in the control design parameterized by $q(s)$ satisfying (4) can now be used to satisfy additional time-domain constraints as will be explained using the typical example of constraints on the step response. The Laplace transform of the closed-loop system's response y to a step input ($r(s) = \frac{1}{s}$) (assuming zero initial conditions) is given by

$$y(s) = \frac{1}{s}\frac{b(s)d(s)}{z(s)} = \frac{1}{s}\frac{b(s)d_0(s)}{z(s)} - \frac{1}{s}\frac{a(s)b(s)}{z(s)}q(s). \tag{5}$$

At this point of the control design a *restrictive assumption* was made [3], namely

Assumption 1 *All the assigned poles $-p_1,\dots,-p_n$ are distinct strictly negative rational numbers.*

Using this assumption and $z(s) = \prod_{i=1}^{n}(s+p_i)$ the partial fractional decomposition of (5) leads to

$$y(s) = \sum_{i=0}^{n} \frac{y_i(q)}{s+p_i}, \tag{6}$$

where $p_0 = 0$ and $y_i(q)$, $i = 1,\dots,n$ are appropriate coefficients following from the decomposition, which are influenced by the choice of the design parameter $q(s) = \sum_{i=0}^{d_q} q_i s^i$. The coefficients $y_i(q)$ depend in an *affine* manner on the parameter $q = (q_0, q_1, \dots, q_{d_q})$, which directly follows by comparing (5) and (6), and equating the coefficients of the powers of s in the resulting numerator polynomials (see also (35) below for an example). The corresponding time-domain signal is given by

$$y(t) = \sum_{i=0}^{n} y_i(q)e^{-p_i t}. \tag{7}$$

Let $p_i = \frac{n_i}{d_i}$ be the ratios of the integers n_i and d_i, and let m denote the least common multiple of the denominators such that $p_i = \frac{\bar{p}_i}{m}$ for some positive integers $\bar{p}_i$. This means that the time-domain output signal at time $t \in \mathbb{R}_+ := [0,\infty)$ can now be expressed as the polynomial

$$y(\lambda, q) = \sum_{i=0}^{n} y_i(q)\lambda^{\bar{p}_i} \tag{8}$$

in the indeterminate $\lambda = e^{-t/m}$. Obviously, λ lies in the interval $[0,1]$ as $t \in \mathbb{R}_+$. Suppose that the output $y(t,q)$ of the system needs to be bounded according to

$$y_{\min} \leq y(t,q) \leq y_{\max} \quad \forall\, t \in \mathbb{R}_+. \tag{9}$$

Formulation (9) is equivalent to enforcing the polynomial bound constraints

$$\left\{ \begin{array}{l} P_1(q,\lambda) := y(\lambda,q) - y_{\min} \geq 0 \\ P_2(q,\lambda) := y_{\max} - y(\lambda,q) \geq 0 \end{array} \right. \quad \forall\, \lambda \in [0,1], \tag{10}$$

where P_1 and P_2 are polynomials in both λ and q. This problem is a special case of the following more general problem of minimizing a polynomial with polynomial constraints over a semialgebraic set, for which we need the following definition in which we use the notation $\mathbb{R}[X^n] := \mathbb{R}[x_1,\dots,x_n]$ to denote the set of polynomials in n variables with real coefficients.

Definition 2 *A set $\mathcal{D}$ is called a basic semialgebraic set if it can be described as*

$$\begin{aligned} \mathcal{D} = \{x \in \mathbb{R}^n \mid e_i(x) \geq 0,\ i = 1,\dots,M_e \text{ and} \\ f_j(x) = 0,\ j = 1,\dots,M_f\} \end{aligned} \tag{11}$$

for certain polynomials $e_i \in \mathbb{R}[X^n]$ and $f_j \in \mathbb{R}[X^n]$.

Problem 3 (Polynomial optimization problem) *Consider two variables $z \in \mathbb{R}^{n_z}$ and $x \in \mathbb{R}^{n_x}$ and let polynomials $g_i \in \mathbb{R}[X^{n_z+n_x}]$, $i = 1,\dots,M_g$, and $p \in \mathbb{R}[X^{n_z}]$ be given. Moreover, let a collection of semialgebraic sets $\mathcal{D}_l \subseteq \mathbb{R}^{n_x}$, $l = 0,\dots,N$ be given. A polynomial optimization problem according to this data is given by*

$$\begin{aligned} &\min_{z} \quad p(z) \\ &\text{s.t.} \quad g_i(z,x) \geq 0,\ i = 1,\dots,M_g\ \forall\, x \in \bigcup_{l=0}^{N} \mathcal{D}_l. \end{aligned} \tag{12}$$

Indeed, (10) can now be written in the form of Problem 3 by taking $z = q$, $x = \lambda$, $M_g = 2$, $N = 1$, $p(z) = 0$, $g_1(z,x) = P_1(q,\lambda)$, $g_2(z,x) = P_2(q,\lambda)$, and $\mathcal{D}_1 = \{\lambda \in \mathbb{R} \mid 0 \leq \lambda \leq 1\}$. Although the bounds $y_{\min}$ and $y_{\max}$ in (10) are chosen to be constants for illustrating purposes, they can also be selected as polynomials in λ, i.e., in the form $y_{\min}(\lambda)$ and $y_{\max}(\lambda)$ without any complications. In this case the bounds in (9) become time-varying. It is well known that by using sum-of-squares techniques [8], [11] these polynomial problems can be solved efficiently using LMI techniques (at least arbitrarily close). As for LMIs there are efficient solvers available, e.g. [12], transforming the problem at hand into Problem 3 provides an effective solution.

III. Problem formulation: the complex poles case

The polynomial representation (7), as derived in [3], of the time response of a linear system to a Laplace transformable input is unfortunately only possible when strictly negative rational closed-loop poles are assigned (see Assumption 1). However, in many cases the assignment of purely real poles can be undesirable, especially in lightly damped systems such as most motion systems. The main objective of this paper is to present a solution to the linear control design problem of time-domain constrained systems of which the Laplace transforms of the closed-loop responses may contain complex roots.

When we allow both distinct real and complex poles to be present in the closed-loop transfer function $T(s)$ and/or the Laplace transformed reference signal $r(s)$, the Laplace transform of the system's output can be decomposed as the partial fractional decomposition

$$y(s)=\sum_{i=0}^{n_r}\frac{y_i}{s+p_i}+\sum_{i=n_r+1}^{n_r+n_c/2+1}\frac{y_i}{s+\alpha_i+j\beta_i}+\frac{y_i^*}{s+\alpha_i-j\beta_i}, \quad (13)$$

where n_r and n_c denote the number of real and complex poles, respectively, $-p_i$ is the location of a real pole, $-\alpha_i \pm j\beta_i$ are the locations of a complex conjugate pair of poles, and y_i are the possibly complex coefficients (with complex conjugate y_i^*) that affinely depend on the design parameter q (we omitted this dependence on q for ease of exposition). To enforce stability, we again assume that the assigned closed-loop poles have strictly negative real part. The corresponding time-domain signal is then described by

$$y(t)=\sum_{i=0}^{n_r} y_i e^{-p_i t} \quad +\sum_{i=n_r+1}^{n_r+n_c/2+1}(y_i e^{-j\beta_i t}+y_i^* e^{j\beta_i t})e^{-\alpha_i t}. \quad (14)$$

As before, we use the following assumption

Assumption 4 *p_i, α_i, and β_i are rational numbers.*

We denote $p_i=\frac{\bar{p}_i}{m}$, $\alpha_i=\frac{\bar{\alpha}_i}{m}$, $\beta_i=\frac{\bar{\beta}_i}{m}$, $\tau=\frac{t}{m}$ for a number m such that $\bar{p}_i$, $\bar{\alpha}_i$, and $\bar{\beta}_i$ can be taken as integers. Furthermore, let $\lambda=e^{-\tau}$. Using Euler's formula $e^{j\phi}=\cos(\phi)+j\sin(\phi)$ and decomposing the complex coefficients as $y_i=a_i+jb_i$, $y_i^*=a_i-jb_i$, yields

$$y(t)=\sum_{i=0}^{n_r} y_i\lambda^{\overline{p}_i}+\sum_{i=n_r+1}^{n_r+n_c/2+1}\left(a_i 2\cos(\overline{\beta}_i\tau)+b_i 2\sin(\overline{\beta}_i\tau)\right)\lambda^{\overline{\alpha}_i}. \quad (15)$$

Obviously, the terms involving the complex poles are non-polynomial in the indeterminate λ because of the presence of products of $\cos(\overline{\beta}_i\tau)$ and $\sin(\overline{\beta}_i\tau)$ with $\lambda^{\overline{\alpha}_i}$, which make it impossible to directly use the positive polynomial approach to bound the output as in (9). Although the parameters $\overline{\alpha}_i$ and $\overline{\beta}_i$ are fixed as a result of the pole placement, there still is freedom in the choice for the coefficients a_i, b_i, which depend on the coefficients $q=(q_0,\ldots,q_{d_q})$ in the Youla-Kučera parameter $q(s)$. We propose a multivariate polynomial relaxation to determine the values y_i, a_i, b_i to shape the time response $y(t)$ that leads to polynomial problems of the type as in Problem 3, which can be chosen to be arbitrarily close to the original constrained problem given by (15),(9) and can be solved using LMIs.

The time response (15) is equivalent to

$$\begin{aligned}y(t)=\sum_{i=0}^{n_r} y_i\lambda^{\overline{p}_i}+&\sum_{i=n_r+1}^{n_r+n_c/2+1}\left[(a_i+jb_i)\left(\cos(\overline{\beta}_i\tau)-j\sin(\overline{\beta}_i\tau)\right)\right.\\ &\left.+(a_i-jb_i)\left(\cos(\overline{\beta}_i\tau)+j\sin(\overline{\beta}_i\tau)\right)\right]\lambda^{\overline{\alpha}_i}.\end{aligned} \quad (16)$$

De Moivre's formula, which is closely related to Euler's formula and $(e^{j\phi})^n=e^{jn\phi}$, states that for any $\phi\in\mathbb{R}$ and any integer $n\in\mathbb{Z}$

$$(\cos(\phi)+j\sin(\phi))^n=\cos(n\phi)+j\sin(n\phi), \quad (17)$$

and hence (16) is equal to

$$\begin{aligned}y(t)=\sum_{i=0}^{n_r} y_i\lambda^{\overline{p}_i}+&\sum_{i=n_r+1}^{n_r+n_c/2+1}\Big((a_i+jb_i)\left[\cos(\tau)-j\sin(\tau)\right]^{\overline{\beta}_i}\\ &+(a_i-jb_i)\left[\cos(\tau)+j\sin(\tau)\right]^{\overline{\beta}_i}\Big)\lambda^{\overline{\alpha}_i}.\end{aligned} \quad (18)$$

Appropriate polynomial functions $w_i:\mathbb{R}^2\to\mathbb{R}$ and $r_i:\mathbb{R}^2\to\mathbb{R}$, $i=n_r+1,\ldots,n_r+n_c/2+1$ in two variables can now be defined such that (18), and thus the time response (15), can be written as

$$\begin{aligned}y(t)=&\sum_{i=0}^{n_r} y_i\lambda^{\overline{p}_i}+\\ &\sum_{i=n_r+1}^{n_r+n_c/2+1}(a_i 2w_i(\cos(\tau),\sin(\tau))+b_i 2r_i(\cos(\tau),\sin(\tau)))\,\lambda^{\overline{\alpha}_i}.\end{aligned} \quad (19)$$

This proves the following theorem.

Theorem 5 *Consider the closed-loop system* (1) *and let y be the response to a reference input r and assume that the Laplace transform $y(s)$ of y has only distinct poles such that* (13) *and Assumption 4 hold. Then we have that*

$$\{y(t)\mid t\in[0,\infty)\}=\{y(u,v,\lambda)\mid(u,v,\lambda)\in\mathcal{F}_{\mathit{original}}\}, \quad (20)$$

where $y(u,v,\lambda)$ is given by the multivariate polynomial

$$y(u,v,\lambda)=\sum_{i=0}^{n_r} y_i\lambda^{\overline{p}_i}+\sum_{i=n_r+1}^{n_r+n_c/2+1}(a_i 2w_i(u,v)+b_i 2r_i(u,v))\,\lambda^{\overline{\alpha}_i}, \quad (21)$$

and

$$\begin{aligned}\mathcal{F}_{\mathit{original}}:=\big\{&(u,v,\lambda)\in\mathbb{R}^3\mid u=\cos(\tau),\ v=\sin(\tau),\\ &\lambda=e^{-\tau}\ \textit{for some}\ \tau\in[0,\infty)\big\}.\end{aligned} \quad (22)$$

Bounding the output as in (9) to the interval $[y_{\min}, y_{\max}]$ is therefore equivalent to enforcing the polynomial non-negativity constraints

$$\begin{aligned}P_3(q,u,v,\lambda)&:=y(u,v,\lambda)-y_{\min}\geq 0,\\ P_4(q,u,v,\lambda)&:=y_{\max}-y(u,v,\lambda)\geq 0\end{aligned} \quad (23)$$

for all $(u,v,\lambda)\in\mathcal{F}_{\text{original}}$. Note that $y(u,v,\lambda)$ depends on q via y_i, a_i, b_i. As we mentioned before, it is of interest to transform the linear constrained control problem into

Problem 3. The conditions (23) are not in this form due to the fact that $\mathcal{F}_{\text{original}}$ is not a (finite union of a) basic semialgebraic set as in Definition 2. However, this set can be overapproximated by a finite union of semialgebaic sets in an arbitrarily close manner.

Definition 6 *We call a set $\mathcal{F}_{approx}$ an ε-*close overapproximation *of $\mathcal{F}_{original}$ for some $\varepsilon > 0$, if it satisfies the following three properties:*

1) *$\mathcal{F}_{approx} = \bigcup_{l=0}^{N} \mathcal{F}_l$ for a finite collection of semialgebraic sets $\mathcal{F}_0, \ldots, \mathcal{F}_N$;*
2) *$\mathcal{F}_{original} \subseteq \mathcal{F}_{approx}$;*
3) *$\mathcal{F}_{approx} \subseteq \mathcal{F}_{original} + \mathbb{B}_\varepsilon$, where $\mathbb{B}_\varepsilon := \{(0,0,z) \mid -\varepsilon \leq z \leq \varepsilon$.*

Hence, an ε-close overapproximation of $\mathcal{F}_{\text{original}}$ contains the set $\mathcal{F}_{\text{original}}$ as drawn by the white line in Figure 2, but it is ε-close in the sense of property 3. Hence, for small $\varepsilon > 0$, replacing $\mathcal{F}_{\text{original}}$ by $\mathcal{F}_{\text{approx}}$ only results in small errors and all guarantees on $\mathcal{F}_{\text{approx}}$ also apply to $\mathcal{F}_{\text{original}}$ due to property 2. Moreover, due to property 1 an ε-close overapproximation $\mathcal{F}_{\text{approx}}$ of $\mathcal{F}_{\text{original}}$ results in a version of Problem 3, where $\mathcal{D}_l = \mathcal{F}_l,\ l = 0, \ldots, N$.

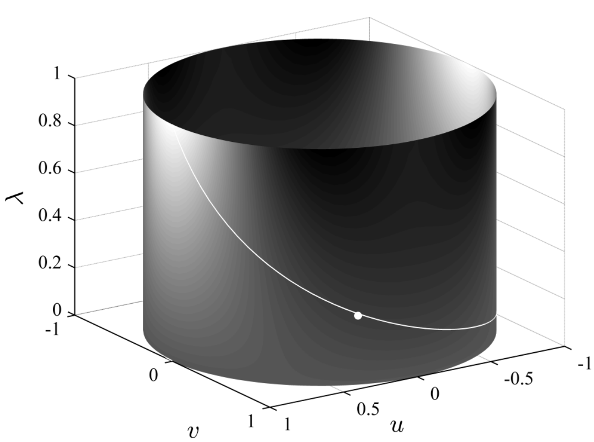

Fig. 2. $\mathcal{F}_{\text{original}}$ (white line) drawn inside the cylinder given by $u^2+v^2 = 1$ and $0 \leq \lambda \leq 1$.

The following algorithm provides an algorithm that constructs for each desirable level of approximation ε an ε-close overapproximation of $\mathcal{F}_{\text{original}}$. The basic idea of the algorithm is to overapproximate the $\mathcal{F}_{\text{original}}$-set by the union of semialgebraic sets $\mathcal{F}_l$, which are obtained by splitting the set $\mathcal{F}_{\text{original}}$ in the τ-direction by considering intervals $I_l := [\tau_l, \tau_{l+1}),\ l = 0, \ldots, N$, where $0 = \tau_0 < \tau_1 < \ldots < \tau_{N+1} = \infty$. On each of these subintervals I_l we approximate $e^{-\tau}$ by $\psi_l(\cos\tau, \sin\tau)$, where $\psi_l : \mathbb{R}^2 \to \mathbb{R}$ is a polynomial such that $|e^{-\tau} - \psi_l(\cos\tau, \sin\tau)| \leq \varepsilon$ for all $\tau \in I_l$. Next to ε, the algorithm uses another parameter $0 < T < 2\pi$, which indicates the desired length of the intervals I_l, $l = 0, \ldots, N-1$ (although it will be modified such that all intervals have the same length).

Algorithm 1 *Let $0 < \varepsilon < 1$ and $0 < T < 2\pi$ be given.*

Step 1: *Define $N := \lceil \frac{-\ln\varepsilon}{T} \rceil$ and $\tau_N := -\ln\varepsilon$ and $\tau_{N+1} := \infty$.*

$$\mathcal{F}_N := \{(u, v, \lambda) \in \mathbb{R}^3 \mid u^2 + v^2 = 1 \text{ and } 0 \leq \lambda \leq \varepsilon\}. \tag{24}$$

Step 2: *Divide the remaining interval $[0, \tau_N)$ in N subintervals of length $\bar{T} := \frac{\tau_N}{N}$. $I_l := [t_l, t_{l+1})$ with $t_l = l\bar{T}$, $l = 0, \ldots, N-1$.*

Step 3: *For each $l = 0, \ldots, N-1$ define a function $\phi_l : \mathbb{R} \to \mathbb{R}$ that satisfies:*

- *ϕ_l is at least continuous, but preferably m times continuously differentiable (C^m) for $m \in \mathbb{N}$ large;*
- *ϕ_l is periodic with period 2π;*
- *$\phi_l(\tau) = e^{-\tau}$ for all $\tau \in I_l$.*

Step 4: *For each $l = 0, \ldots, N-1$ use the Fourier series to obtain an approximation of ϕ_l in the sense that*

$$|\phi_l(\tau) - \sum_{k=0}^{K_l} [a_k \cos(k\tau) + b_k \sin(k\tau)]| \leq \varepsilon \text{ for all } \tau \in I_l \tag{25}$$

with $a_k, b_k,\ k = 0, \ldots, K_l$ the Fourier coefficients of ϕ_l.

Step 5: *For each $l = 0, \ldots, N-1$ use De Moivre's formula to rewrite $\sum_{k=0}^{K_l} [a_k \cos(k\tau) + b_k \sin(k\tau)]$ obtained in the previous step as*

$$\sum_{k=0}^{K_l} \sum_{i=0}^{k} c_{ki} (\cos(\tau))^k (\sin(\tau))^l =: \psi_l(\cos(\tau), \sin(\tau)),$$

where $\psi_l : \mathbb{R}^2 \to \mathbb{R}$ is a polynomial.

Step 6: *For each $l = 0, \ldots, N-1$, define*

$$\begin{aligned} \mathcal{F}_l := \{(u, v, \lambda) \in \mathbb{R}^3 \mid -\varepsilon \leq \lambda - \psi_l(u, v) \leq \varepsilon \wedge \\ u^2 + v^2 = 1 \wedge (S_l - S_{l+1})u + (C_{l+1} - C_l)v + \\ S_{l+1}C_l - C_{l+1}S_l \leq 0\}, \end{aligned} \tag{26}$$

where $C_l := \cos(t_l),\ S_l := \sin(t_l)$.

Step 7: *Take $\mathcal{F}_{approx} = \bigcup_{l=0}^{N} \mathcal{F}_l$.*

Theorem 7 *For each $0 < \varepsilon < 1$ and $0 < T < 2\pi$ Algorithm 1 produces an ε-close overapproximation $\mathcal{F}_{approx}$ of $\mathcal{F}_{original}$.*

For brevity the proof is omitted. Using the ε-close overapproximation $\mathcal{F}_{\text{approx}} = \bigcup_{l=0}^{N} \mathcal{F}_l$, the conditions (23) for all $(u, v, \lambda) \in \mathcal{F}_{\text{approx}}$ can be written as a special case of (12) with $z = q,\ x = (u, v, \lambda),\ \ell = 2,\ m = 0,\ p(z, x) = 0,\ g_1(z, x) = P_5(q, u, v, \lambda),\ g_2(z, x) = P_6(q, u, v, \lambda)$, and $\mathcal{D}_l = \mathcal{F}_l,\ l = 0, \ldots, N$.

For $\varepsilon = e^{-1.5\pi} \approx 0.009$ and $T = \bar{T} = 0.75\pi$ the ε-close overapproximation $\mathcal{F}_{\text{approx}}$ of $\mathcal{F}_{\text{original}}$ can be generated with $\tau_0 = 0$, $\tau_1 = 0.75\pi$, $\tau_2 = 1.5\pi$ and $\tau_3 = \infty$ and polynomials

$$\begin{aligned} \psi_0(u, v) &= 0.398u - 0.971v + 0.616u^2 - 0.192uv + 1.179v^2 \\ &\quad - 0.015u^3 + 0.184u^2 v, \\ \psi_1(u, v) &= 0.033u + 0.096v + 0.0760u^2 + 0.0534uv + 0.094v^2 \\ &\quad + 0.013uv^2 - 0.011v^3. \end{aligned} \tag{27}$$

This overapproximation and the polynomials are illustrated in Figure 3. If one is satisfied with an overapproximation

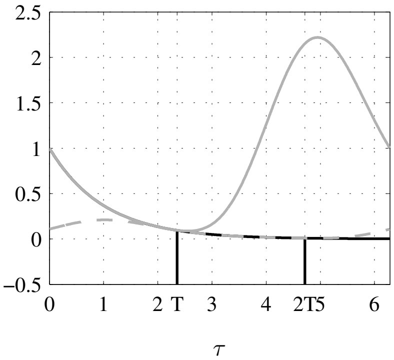

Fig. 3. Functions $e^{-\tau}$ (solid black), ψ_0 (solid grey), and ψ_1 (dashed grey).

accuracy of $\varepsilon = e^{-1.5\pi} \approx 0.009$, then one can use this precomputed overapproximation. If it is desired to have a simpler overapproximation (with less regions and polynomials of lower degree) or an even tighter approximations $\varepsilon < 0.009$, one can run Algorithm 1 to obtain it.

The objective function $p(z, x)$ of Problem 3 can be used to express additional desired properties of the response (such as the steady-state error and overshoot) in terms of the design parameters y_i, a_i, and b_i. Consider the step response for example, which can be written as (13) with $p_0 = 0$ and where y_0 is the steady-state solution. Desirable properties of the unit step response are, for instance, a zero steady-state error, a small settling-time and small overshoot. These properties can be accommodated in $p(z)$ as follows

- **small steady-state error:** Set $p(z) = (1 - y_0)^2$ to minimize the steady-state error.
- **overshoot minimization:** Constrain the response (21) as $y(u, v, \lambda) \leq \gamma$ and specify $p(z) = \gamma$ to minimize the overshoot.
- **fast settling:** Set $p(z) = a_i^2 + b_i^2$, where index i corresponds to a slow mode in (15), to minimize the contribution of this mode, which improves the settling time. Alternatively, exponentially decreasing constraints can be specified that directly impose a certain desired settling behavior.

The proposed control design procedure will be illustrated by an example in the next section.

IV. EXAMPLE

The efficiency of the proposed design method will be demonstrated by means of a simulation example. Consider the simple model given by

$$P(s) = \frac{y(s)}{u(s)} = \frac{1}{s+1}. \tag{28}$$

The control objective is to let y track a step reference from 0 to 1 as close as possible. Moreover, the controller will be designed such that the assigned complex closed-loop poles are $p_{1,2} = -1 \pm 2j$, $p_{3,4} = -2 \pm 4j$. This is done by solving the Diophantine equation (2) leading to the d-minimal controller

$$C(s) = \frac{d_0(s)}{c_0(s)} = \frac{68}{s^3 + 5s^2 + 28s + 32}, \tag{29}$$

resulting in the closed-loop system given by the complementary sensitivity function

$$T(s) = \frac{68}{s^4 + 6s^3 + 33s^2 + 60s + 100}. \tag{30}$$

Using the Youla-Kučera parameter $q(s)$ and realizing that according to (4) we have $\deg q(s) \leq 2$, i.e., $q(s) = q_0 + q_1 s + q_2 s^2$, the set of allowable controllers assigning the specified closed-loop poles is parameterized as

$$\begin{aligned} C(s) &= \frac{d(s)}{c(s)} = \frac{d_0(s) - a(s)q(s)}{c_0(s) + b(s)q(s)} \\ &= \frac{68 - (q_0 + (q_0 + q_1)s + (q_1 + q_2)s^2 + q_2 s^3)}{s^3 + 5s^2 + 28s + 32 + (q_0 + q_1 s + q_2 s^2)}, \end{aligned} \tag{31}$$

resulting in the set of closed-loop transfer functions

$$T(s) = \frac{68 - (q_0 + (q_0 + q_1)s + (q_1 + q_2)s^2 + q_2 s^3)}{s^4 + 6s^3 + 33s^2 + 60s + 100}. \tag{32}$$

The Laplace transform of the time response of (32) to a step input is then parameterized as

$$\begin{aligned} y(s) &= \frac{1}{s}T(s) = \\ &\frac{68 - (q_0 + (q_0 + q_1)s + (q_1 + q_2)s^2 + q_2 s^3)}{s(s+1+2j)(s+1-2j)(s+2+4j)(s+2-4j)}. \end{aligned} \tag{33}$$

The corresponding partial fractional decomposition is equal to

$$y(s) = \frac{y_0}{s} + \frac{a_1 + jb_1}{s+1+2j} + \frac{a_1 - jb_1}{s+1-2j} + \frac{a_2 + jb_2}{s+2+4j} + \frac{a_2 - jb_2}{s+2-4j} \tag{34}$$

where y_0, a_1, b_1, a_2, b_2 can be solved from the linear system of equations

$$\begin{bmatrix} 100 & 0 & 0 & 0 & 0 \\ 60 & 40 & 80 & 20 & 40 \\ 33 & 48 & 16 & 18 & 16 \\ 6 & 10 & 4 & 8 & 8 \\ 1 & 2 & 0 & 2 & 0 \end{bmatrix} \begin{bmatrix} y_0 \\ a_1 \\ b_1 \\ a_2 \\ b_2 \end{bmatrix} = \begin{bmatrix} 68 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -1 & 0 & 0 \\ -1 & -1 & 0 \\ 0 & -1 & -1 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \end{bmatrix}, \tag{35}$$

where q_0, q_1, q_2 are the free variables in the Youla-Kučera parameter to shape the time response. The goal is to determine values of y_0, a_1, b_1, a_2, b_2 (via q_0, q_1, q_2) such that the closed-loop time response to the step input can be shaped. We have that $m = 1, n_r = 0, n_c = 2$, $\overline{\alpha}_1 = -1$, $\overline{\alpha}_2 = -2$, $\overline{\beta}_1 = 2$ and $\overline{\beta}_2 = 4$. Let $u = \cos(\tau)$ and $v = \sin(\tau)$ so that (18) yields

$$\begin{aligned} y(t) &= \Big((a_1 + jb_1)(u + jv)^2 + (a_1 - jb_1)(u + jv)^2 \Big) \lambda \\ &+ \Big((a_2 + jb_2)(u + jv)^4 + (a_2 - jb_2)(u + jv)^4 \Big) \lambda^2 \\ &= \left(2a_1 \left(u^2 - v^2 \right) + 2b_1 2uv \right) \lambda \\ &+ \left(2a_2 \left(u^4 + v^4 - 6u^2 v^2 \right) + 2b_1 \left(4vu^3 - 4uv^3 \right) \right) \lambda^2. \end{aligned} \tag{36}$$

As a consequence, for this example we obtain

$$\begin{aligned} w_1(u,v) &= u^2 - v^2, \quad r_1(u,v) = 2uv \\ w_2(u,v) &= u^4 + v^4 - 6u^2v^2, \quad r_2(u,v) = 4vu^3 - 4uv^3. \end{aligned} \tag{37}$$

yielding the time response

$$\begin{aligned} y(u,v,\lambda) =& y_0 + (2a_1(u^2 - v^2) + 4b_1uv)\lambda \\ &+ (2a_2(u^4 + v^4 - 6u^2v^2) + 8b_2(vu^3 - uv^3))\lambda^2, \end{aligned} \tag{38}$$

which is a multivariate polynomial with 3 independent variables (u, v, λ) and three decision variables (q_0, q_1, q_2). Note that $(u, v, \lambda) \in \mathcal{F}_{\text{original}}$. Since $\mathcal{F}_{\text{original}}$ is not the finite union of a basic semialgebraic set, we can use Algorithm 1 or use the precomputed overapproximation from Section III to obtain an ε-close overapproximation $\mathcal{F}_{\text{approx}}$ of $\mathcal{F}_{\text{original}}$. We use here the precomputed overapproximation $\mathcal{F}_{\text{approx}}$ with $\varepsilon = e^{-1.5\pi} \approx 0.009$. In accordance with Section III, we formulate the problem as to find $q(s)$ such that the overshoot γ is small and that the steady-state error of the step response is minimized. Therefore, the problem is posed as

$$\begin{aligned} \min_{q_0,q_1,q_2} \quad & 10(1 - y_0)^2 + \gamma \\ \text{s.t.} \quad & (35) \\ & \gamma - y(u,v,\lambda) \geq 0 \quad \forall (u,v,\lambda) \in \mathcal{F}_{\text{approx}}. \end{aligned} \tag{39}$$

Rewriting this optimization problem gives

$$\begin{aligned} \min_{q_0,q_1,q_2} \quad & 10(1 - y_0)^2 + \gamma \\ \text{s.t.} \quad & (35) \\ & \gamma - y(u,v,\lambda) \geq 0 \quad \forall (u,v,\lambda) \in \mathcal{F}_0 \\ & \gamma - y(u,v,\lambda) \geq 0 \quad \forall (u,v,\lambda) \in \mathcal{F}_1 \\ & \gamma - y(u,v,\lambda) \geq 0 \quad \forall (u,v,\lambda) \in \mathcal{F}_2, \end{aligned} \tag{40}$$

with $\mathcal{F}_{0,1}$ as in (26) and $\mathcal{F}_2$ as in (24). The resulting $\tilde{\gamma}$ is equal to 1.0718, representing an overshoot of 7.18%. The corresponding Youla-Kučera parameter is given by

$$q(s) = -32.0 - 17.0607s - 3.0227s^2, \tag{41}$$

which yields the controller

$$C(s) = \frac{3.0s^3 + 20.0s^2 + 49s + 100}{s^3 + 2.0s^2 + 10.9s}. \tag{42}$$

The step responses of both the original closed-loop with the d-minimal controller (29) and of the closed-loop with controller (42) are depicted in Fig. 4, which shows a significant improvement as expected. The maximum of the step response $y(t)$ equals 1.0714 (i.e., 7.14% overshoot) which indeed is ε-close to $\tilde{\gamma} = 1.0718$ and indicates that $\tilde{\gamma}$ in fact is close to the global minimum.

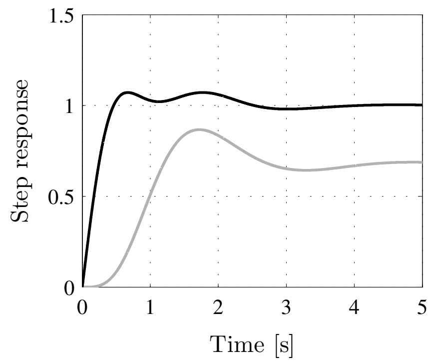

Fig. 4. New (black) and original (grey) step responses.

V. Conclusions

Recent results on the control of linear systems subject to time-domain constraints were only applicable to the case of real closed-loop poles, which is in various practical situations a severe restriction. In this paper we removed this restriction and proposed a framework for the design of a controller subject to closed-loop time-domain constraints, also in case that complex poles are present. The controllers are synthesized via a closed-loop pole placement method in which the additional design freedom in terms of the Youla-Kučera parameter is used to satisfy time-domain constraints. These constraints are reformulated as LMIs using goniometrical ideas combined with relaxations of multivariate polynomial optimization problems over semialgebraic sets. We proved that these relaxations can approximate the original problem with arbitrary accuracy. In addition, we showed how important closed-loop properties such as overshoot and steady-state error can effectively be reduced using this design framework.

VI. Acknowledgments

The authors are grateful to Prof. Didier Henrion for his helpful comments and suggestions.

References

[1] J. Doyle, B. Francis, and A. Tannenbaum, *Feedback control theory*. New York: MacMillan, 1992.
[2] G. Franklin, J. Powell, and A. Emami-Naeini, *Feedback control of dynamic systems*. New Jersey: Prentice Hall, 2002.
[3] D. Henrion, S. Tarbouriech, and V. Kucera, "Control of linear systems subject to time-domain constraints with polynomial pole placement and LMIs," *Trans. Aut. Contr.*, vol. 50, no. 9, pp. 1360–1364, 2005.
[4] V. Kucera, "The pole placement equation - a survey," *Kybernetica*, vol. 30, pp. 578–584, 1994.
[5] Y. Nesterov, "Squared functional systems and optimization problems," in *High performance optimization*, H. Frenk, K. Roos, T. Terlaky, and S. Zhang, Eds. Dordrecht, The Netherlands: Kluwer academic publishers, 2000, ch. 17, pp. 405–440.
[6] D. Henrion and J.-B. Lasserre, "LMIs for constrained polynomial interpolation with application in trajectory planning," *System & Control Letters*, vol. 55, no. 6, pp. 473–477, 2006.
[7] A. Papachristodoulou and S. Prajna, "A tutorial on sum of squares techniques for systems analysis," in *Proceedings of the American Control Conference*, vol. 4, 2005, pp. 2686–2700.
[8] P. Parrilo, "Structured semidefinte programs and semialgebraic geometry methods in robustness and optimization," Ph.D. dissertation, California Institute of Technology, Pasadena, CA, 2000.
[9] B. Francis, *A course in H-infinity control theory*, ser. Lecture Notes in Control and Information Science. Springer, 1987, vol. 88.
[10] V. Kucera and P. Zagalak, "Proper solutions of polynomial equations," in *14th IFAC World Congress*, Beijing, China, 1999, pp. 357–362.
[11] J. Lasserre, "Global optimization with polynomials and the problem of moments," *SIAM J. on Optimization*, vol. 11, pp. 796–817, 2001.
[12] J. Sturm, "Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones," *Optimization methods and software*, vol. 11-12, pp. 625–653, 1999.