Preprints of the 20th World Congress
The International Federation of Automatic Control
Toulouse, France, July 9-14, 2017

Tube-based LPV constant output reference tracking MPC with error bound ⋆

Jurre Hanema *, **Mircea Lazar** *, **Roland Tóth** *

* *Eindhoven University of Technology, P.O. Box 513, 5600 MB, Eindhoven, The Netherlands ({j.hanema,m.lazar,r.toth}@tue.nl).*

Abstract: We address the constrained output tracking of constant references for linear parameter-varying systems. In model predictive control based on linear parameter-varying state-space representations, offset-free output tracking is generally not possible due to the variations in the scheduling signal. Therefore, we aim instead to guarantee a pre-specified tracking error bound which is achievable for all admissible variations of the scheduling variable. The construction of an invariant set in which such a bound can be satisfied is described. Subsequently a tube-based model predictive controller is designed which brings the state of the system inside this set in finite time. The properties of the approach are demonstrated on numerical examples.

Keywords: Predictive control, linear parameter-varying systems, reference tracking

1. INTRODUCTION

Linear parameter-varying (LPV) systems can be used to model applications exhibiting operating point-dependent behavior. In an LPV system, the dynamical mapping between inputs and outputs is linear while the mapping itself depends on a time-varying and measurable *scheduling variable*, denoted by θ. Applications of LPV modeling and control can be found, e.g., in the automotive, aerospace and mechatronic domains (Mohammadpour and Scherer, 2012; Hoffmann and Werner, 2014). When it is desired to control an LPV system that is subject to constraints on its inputs, outputs, or state variables, the application of *model predictive control* (MPC) becomes attractive.

The design of stabilizing MPC laws for LPV systems has been studied extensively in the literature. Early approaches such as (Lu and Arkun, 2000; Casavola et al., 2002) were based on the on-line synthesis of linear controllers by semidefinite programming. More recent designs apply the principles of tube MPC (*TMPC*) (Langson et al., 2004) to LPV systems: examples include, e.g., (Fleming et al., 2015; Muñoz-Carpintero et al., 2015; Hanema et al., 2016). Typically, these tube-based methods require the on-line solution of quadratic- or linear programs.

Besides stabilization of the state around the origin, it is often desired to bring the *output* of the system to a specified value leading to a reference tracking problem. Related to the LPV case, robust tracking MPC approaches for uncertain linear time-varying (LTV) systems were developed in, e.g., (Pannocchia, 2004; Wang and Rawlings, 2004). It is shown that offset-free tracking is achieved if the uncertainty (corresponding to θ in an LPV system) is time-invariant. However, no hard bounds on the tracking error have been given for the general case when θ is a time-varying signal.

When controlling a constrained LPV system described by a state-space representation, it is usually impossible to achieve offset-free output tracking even of constant reference signals because variations in θ also influence the output. Indeed, in MPC, offset-free tracking can generally only be achieved for (asymptotically) constant disturbances or uncertainties (Pannocchia et al., 2015). Then, it becomes a natural approach to look for a set *around* the desired reference to which convergence can be established.

Previously, in the LTI case, (Alvarado et al., 2007) considered the tracking of piecewise constant references for LTI systems subject to additive disturbances. Using a tube-based approach, asymptotic convergence of the output towards a bounded region around the reference is established. The paper (Betti et al., 2013) presents similar results using prediction models in the velocity form. In (Falugi and Mayne, 2013), undisturbed LTI systems with randomly time-varying references are considered: again, a bounded set to which the tracking error converges is characterized.

An alternative form of LTI tracking MPC guaranteeing strict error bounds was recently proposed (Di Cairano and Borrelli, 2016). The reference is assumed to be generated by a *reference generator*, which itself is a constrained LTI system subject to bounded additive disturbances. An invariant set is designed in a lifted state/reference space. When the state of the system and the state of the reference generator are inside this set, the tracking error is contained within an ϵ-ball for all possible reference trajectories. An MPC law is designed which brings the state of the system inside this set, provided that the initial tracking error is already within the specified ϵ-ball.

To resolve the MPC tracking problem for LPV systems, we adopt the basic reasoning of (Di Cairano and Borrelli, 2016) and modify it to handle the LPV setting. Recognizing that offset-free tracking is generally impossible when θ is time-varying, we derive an invariant set in which the tracking error with respect to a given *constant* reference satisfies a pre-specified bound. Then, a tube-based predictive controller based on (Hanema et al., 2016) is designed

⋆ This research was supported by the Impulse 1 program of the Eindhoven University of Technology and ASML.

to control the state of the system towards this set. We employ a time-varying terminal constraint to ensure finite-time convergence, while the finite-horizon cost function is designed to optimize local tracking performance. In this proposed method, which serves as a stepping stone towards the development of more sophisticated constrained LPV output reference tracking controllers, we only consider constant references in lieu of the more general signal class from (Di Cairano and Borrelli, 2016). The controller proposed in this paper does not require the initial tracking error to be contained already inside of the ϵ-ball.

The organization of the paper is as follows. In Section 2 all preliminaries including notation, a formal problem definition, and an introduction to LPV tube MPC, are given. In Section 3, the TMPC algorithm is adapted to the tracking case. The approach is demonstrated on two numerical examples in Section 4 and the paper concludes with some remarks and ideas for future improvements.

2. PRELIMINARIES

2.1 Notation and basic definitions

The symbol $\mathbb{R}_+$ denotes the nonnegative real numbers and $\mathbb{N}$ denotes the nonnegative integers including zero. Define a closed index set as $\mathbb{N}_{[a,b]} := \{i \in \mathbb{N} \mid a \leq i \leq b\}$. In this paper, a norm without explicit subscript $\|x\|$ refers to the ∞-norm of a vector $x \in \mathbb{R}^n$, i.e., $\|x\| = \|x\|_\infty = \max_{i\in\mathbb{N}_{[1,n]}} |x_i|$. For a set $Y \subseteq \mathbb{R}^n$, a scalar $\alpha \in \mathbb{R}$ and a vector $v \in \mathbb{R}^n$, let $\alpha Y = \{\alpha y \mid y \in Y\}$, $v \oplus Y = \{y + v \mid y \in Y\}$, and let $\mathrm{convh}\{Y\}$ be the convex hull of Y. A subset of $\mathbb{R}^n$ is a polytope if it is compact and an intersection of finitely many half-spaces or, equivalently, the convex hull of finitely many points in $\mathbb{R}^n$. A convex and compact set $X \subset \mathbb{R}^n$ which contains the origin in its non-empty interior is called a proper C-set, or PC-set. The value of a signal $w : \mathbb{N} \to \mathbb{R}^{n_\mathrm{w}}$ at time k is written as $w(k)$. The value of w at time instant $k+i$, predicted from information available up to and including time k, is denoted by $w_{i|k}$. Sequences of predicted variables are compactly denoted by $\{X_{i|k}\}_a^b := \{X_{a|k}, X_{a+1|k}, \ldots, X_{b|k}\}$ for any $b \geq a$. A closed ϵ-ball is defined as

$$\mathcal{B}(\epsilon) := \{x \mid \|x\| \leq \epsilon\}$$

where the dimension of the space will be clear from the context.

2.2 Problem setting

We consider a constrained LPV system, represented by the following state-space equation

$$\begin{aligned} x(k+1) &= A(\theta(k))x(k) + B(\theta(k))u(k) \\ y(k) &= C(\theta(k))x(k) \end{aligned} \tag{1}$$

with $x(0) = x_0$, and where $u : \mathbb{N} \to \mathcal{U} \subseteq \mathbb{R}^{n_\mathrm{u}}$ is the input, $y : \mathbb{N} \to \mathcal{Y} \subseteq \mathbb{R}^{n_\mathrm{y}}$ is the output, $x : \mathbb{N} \to \mathcal{X} \subseteq \mathbb{R}^{n_\mathrm{x}}$ is the state vector, and $\theta : \mathbb{N} \to \Theta \subseteq \mathbb{R}^{n_\theta}$ is the scheduling signal. The sets $\mathcal{U}$, $\mathcal{Y}$ and $\mathcal{X}$ are the input-, output- and state constraint sets, respectively, while Θ is called the scheduling set. The matrices $A(\theta)$, $B(\theta)$ and $C(\theta)$ in (1) are considered to be real affine functions of θ, i.e.,

$$\begin{bmatrix} A(\theta) & B(\theta) \\ C(\theta) & 0 \end{bmatrix} = \begin{bmatrix} A_0 & B_0 \\ C_0 & 0 \end{bmatrix} + \sum_{i=1}^{n_\theta} \begin{bmatrix} A_i & B_i \\ C_i & 0 \end{bmatrix} \theta_i \tag{2}$$

where (A_i, B_i, C_i), $i \in \mathbb{N}_{[0,n_\theta]}$ are matrices of conformable dimensions. We start with a simplifying assumption.

Assumption 1. The state- and output constraint sets are consistent, i.e., $\forall (x,\theta) \in \mathcal{X} \times \Theta :\ y = C(\theta)x \in \mathcal{Y}$.

Due to this assumption, it is sufficient to consider only the two constraint sets $\mathcal{X}$ and $\mathcal{U}$ explicitly, the output constraint $\mathcal{Y}$ being implicitly satisfied through satisfaction of $\mathcal{X}$. Furthermore, we assume the following.

Assumption 2. (i) The values $x(k)$ and $\theta(k)$ can be measured for all $k \in \mathbb{N}$. (ii) The sets $\mathcal{X}$ and $\mathcal{U}$ are polytopic PC-sets. (iii) The set Θ is a polytope with q vertices, i.e., $\Theta = \mathrm{Co}\{\bar{\theta}^j \mid j \in \mathbb{N}_{[1,q]}\}$. (iv) It holds $1 \leq n_\mathrm{y} \leq n_\mathrm{u}$, and

$$\forall \theta \in \Theta :\ \mathrm{rank} \begin{bmatrix} A(\theta) - I & B(\theta) \\ C(\theta) & 0 \end{bmatrix} = n_\mathrm{x} + n_\mathrm{y}.$$

Assumption 2.(iv) implies that for any reference $r \in \mathcal{Y}$ and any constant scheduling value $\bar{\theta} \in \Theta$, it is possible to find a steady state/input pair $\big(\bar{x}(\bar{\theta}, r), \bar{u}(\bar{\theta}, r)\big)$ such that $C(\bar{\theta})\bar{x}(\bar{\theta}, r) = r$. This will prove to be useful later in the design of a tracking MPC cost function. The following problem is considered in this paper.

Problem 3. Given a reference value $r \in \mathcal{Y}$ and a bound $\epsilon \in \mathbb{R}_+$, design a controller which (i) brings the tracking error $e(k) = y(k) - r$ into the ϵ-ball $\mathcal{B}(\epsilon)$ in finite time, and (ii) subsequently keeps it there forever despite of the variations of $\theta \in \Theta$.

As a final preliminary, the next subsection introduces the basics of the LPV tube MPC algorithm that will be employed and modified to obtain a solution to Problem 3.

2.3 Basic LPV TMPC setup

In this subsection, we review the basic LPV TMPC from (Hanema et al., 2016) which will be adapted to the tracking problem considered in this paper. The algorithm brings the state of the system into a set by constructing on-line, at each sampling instant, a constraint invariant tube:

Definition 4. A constraint invariant tube for the constraint set $(\mathcal{X}, \mathcal{U}) \subseteq \mathbb{R}^{n_\mathrm{x}} \times \mathbb{R}^{n_\mathrm{u}}$ is defined as $\mathbf{T}_k := \big(\mathbf{X}_k, \mathbf{\Pi}_k\big) = \Big(\{X_{i|k}\}_0^N, \{\Pi_{i|k}\}_0^{N-1}\Big)$ where $X_{i|k} \subseteq \mathbb{R}^{n_\mathrm{x}}, i \in \mathbb{N}_{[0,N]}$ are sets and $\Pi_{i|k} : X_{i|k} \times \Theta_{i|k} \to \mathcal{U}, i \in \mathbb{N}_{[0,N-1]}$ are control laws satisfying the condition $\forall (x,\theta) \in X_{i|k} \times \Theta_{i|k} : A(\theta)x + B(\theta)\Pi_{i|k}(x,\theta) \in X_{i+1|k} \cap \mathcal{X}$. The sequence $\mathbf{X}_k$ is called the state tube, and each set $X_{i|k}$ is called a cross section.

For simplicity of the presentation, in this paper, it is assumed that for all $k \in \mathbb{N}$, we have $\Theta_{0|k} = \{\theta(k)\}$ and $\forall i \in \mathbb{N}_{[1,N-1]} : \Theta_{i|k} = \Theta$. However, it is possible to take expected variations of the scheduling variable into account by designing the sets $\Theta_{i|k}$ accordingly (Hanema et al., 2016). To synthesize a constraint invariant tube through the solution of a finite-dimensional optimization problem, the sets and controllers from the above definition both need to be characterized by a finite number of parameters. In this work, we consider the following "parameterized tube"-variety.

Definition 5. Introduce a parameter set $\mathbb{P} = \mathbb{R}^{q_\mathrm{p}^\mathrm{x}} \times \mathbb{R}^{q_\mathrm{p}^\pi}$ where $\big(q_\mathrm{p}^\mathrm{x}, q_\mathrm{p}^\pi\big) \in \mathbb{N}_{[0,N]}^2$. Assume that there is a function $\bar{P}(\cdot)$ mapping these parameters to sets and control laws. A tube $\mathbf{T}_k$ according to Definition 4 is called a parameterized tube if $\forall i \in \mathbb{N}_{[0,N]}$ there exists a $p_{i|k} \in \mathbb{P}$ such that $\left(X_{i|k}, \Pi_{i|k}\right) = \bar{P}\left(p_{i|k}\right)$.

Note that in the above definition, the tube parameters $p_{i|k}$ can be decomposed as $p_{i|k} = \left(p^{\mathrm{x}}_{i|k}, p^{\pi}_{i|k}\right)$ where $p^{\mathrm{x}}_{i|k} \in \mathbb{R}^{q^{\mathrm{x}}_{\mathrm{p}}}$ parameterizes the sets $X_{i|k}$ and $p^{\pi}_{i|k} \in \mathbb{R}^{q^{\pi}_{\mathrm{p}}}$ parameterizes the associated controllers $\Pi_{i|k}(\cdot,\cdot)$. In this paper, we use the same parameterizations as in (Hanema et al., 2016): the cross sections are homothetic to a given set and the control laws are the associated scheduling set-induced vertex controllers. More precisely, we have

$$\forall (k,i) \in \mathbb{N} \times \mathbb{N}_{[0,N]}: \; X_{i|k} = z_{i|k} \oplus \alpha_{i|k} S \tag{3}$$

with S is a polytopic PC-set designed off-line and where $p^{\mathrm{x}}_{i|k} = \left(\alpha_{i|k}, z_{i|k}\right)$ are the cross section parameters optimized on-line. In this work, S will be designed as an invariant set for the system (1) subject to a tracking error constraint: the details are discussed later in Section 3. Since S is a polytope, it can be represented as the convex hull of q_s vertices. We choose $\Pi_{i|k}(\cdot,\cdot)$ to be characterized by the vertex controllers

$$\Pi_{i|k}(x,\theta) = \sum_{j=1}^{q_s} \mu^j_{i|k} \sum_{l=1}^{q} \lambda^l_{i|k} u^{(j,l)}_{i|k}, \tag{4}$$

where $\mu_{i|k} \in \mathbb{R}^{q_s}_+$ and $\lambda_{i|k} \in \mathbb{R}^q_+$ are such that $\sum_{j=1}^{q_s} \mu^j_{i|k} = 1$, $\sum_{j=1}^{q_s} \mu^j_{i|k} \bar{x}^j_{i|k} = x$, $\sum_{l=1}^{q} \lambda^l_{i|k} = 1$, and $\sum_{l=1}^{q} \lambda^l_{i|k} \bar{\theta}^l_{i|k} = \theta$. The convex multipliers $\left(\mu_{i|k}, \lambda_{i|k}\right)$ are never actually computed: only the control actions in $p^{\pi}_{i|k} = \left(u^{(1,1)}_{i|k}, \dots, u^{(q_s,q)}_{i|k}\right)$ are synthesized on-line.

Let $\mathbf{d}_k \in \mathbb{D} = \mathbb{R}^{n_{\mathrm{d}}}$ be a decision variable containing, besides possibly some auxiliary or slack variables, all parameters $p_{i|k}$, $i \in \mathbb{N}_{[0,N]}$ required to fully characterize a tube $\mathbf{T}_k$. The on-line construction then requires solving, at each time instant $k \in \mathbb{N}$, the optimization problem

$$\begin{aligned} V(k, x_{0|k}, \theta_{0|k}) = \min_{\mathbf{d}} \; & J_N(k, \mathbf{d}) \\ \text{s.t. } & \mathbf{d} \in \mathcal{D}_N(k, x_{0|k}, \theta_{0|k}) \end{aligned} \tag{5}$$

where $J_N(k, \mathbf{d})$ is a cost function and $\mathcal{D}_N(k, x, \theta)$ is the set of all decision variables $\mathbf{d}$ characterizing a tube satisfying the constraints of Definition 4 and an additional terminal constraint. As it is usual in MPC, a terminal set constraint is required to obtain recursive feasibility and convergence. Its design will be described in Section 3. As a last important step, we invoke the following assumption to guarantee convexity of (5).

Assumption 6. At least one of the following conditions is satisfied: (i) The input matrix of (1) is constant, i.e., $\forall \theta \in \Theta: \; B(\theta) = B$, or (ii) the controllers $\Pi_{i|k}(\cdot,\cdot)$ from (4) are independent of θ, i.e., $\Pi(x,\theta) = \Pi(x)$.

A system with a parameter-varying input matrix can always be transformed into a system with a constant input matrix by appending a stable input filter or a pre-integrator (Blanchini et al., 2007). In this way, it is possible to synthesize parameter-dependent controllers also for systems with a varying B-matrix.

In the next section, a suitable invariant terminal set is derived. This is then used to construct the set $\mathcal{D}_N(\cdot,\cdot,\cdot)$ and the cost function $J_N(\cdot,\cdot)$ so as to obtain an LPV TMPC synthesis which solves Problem 3.

3. TUBE-BASED LPV TRACKING MPC

In this section, we describe the three necessary components for adapting the LPV TMPC approach described previously to the output reference tracking case. These components are (i) an invariant target set in which a certain tracking error bound is satisfied, (ii) a time-varying terminal constraint yielding finite-time convergence to the target set, and (iii) a cost function designed to optimize tracking performance.

3.1 Bounded-error invariant set

In this subsection, we propose the construction of an invariant set which can be used as a target set in our final MPC strategy.

Definition 7. A set S is called *controlled positively invariant* (CPI) for system (1) with respect to the constraints $\mathcal{X} \times \mathcal{U}$ if $S \subseteq \mathcal{X}$ and $\forall (x,\theta) \in S \times \Theta: \; \exists u \in \mathcal{U}: \; A(\theta)x + B(\theta)u \in S$.

Furthermore, S is called the *maximal CPI* (MCPI) set with respect to $\mathcal{X} \times \mathcal{U}$ if it contains all other sets which are CPI with respect to $\mathcal{X} \times \mathcal{U}$. A polytopic contractive inner-approximation of the maximal invariant set for a system of the form (1) satisfying Assumption 2, can be computed iteratively in a finite number of iterations (Blanchini and Miani, 2008). In light of Assumption 6, it is emphasized that the input u in Definition 7 is allowed to depend on θ only if the input matrix of (1) is constant.

Due to the variations in θ, it is generally impossible to achieve offset-free tracking even for constant references r. Instead, we aim to keep the tracking error

$$e(k) := y(k) - r = C(\theta(k))x(k) - r$$

within a certain bound which can be maintained for all admissible scheduling trajectories. This bound will be characterized in terms of an ϵ-ball $\mathcal{B}(\epsilon)$. It is first assumed that a reference r and a bound $\epsilon \in \mathbb{R}_+$ are given. Later, we describe the case when only a desired error bound is specified a-priori. Define an augmented state constraint set

$$\mathcal{X}_{\mathcal{E}}(r,\epsilon) := \mathcal{X} \cap \mathcal{E}(r,\epsilon)$$

where

$$\mathcal{E}(r,\epsilon) := \{x \in \mathbb{R}^{n_{\mathrm{x}}} \mid \forall \theta \in \Theta: \; (C(\theta)x - r) \in \mathcal{B}(\epsilon)\}$$

contains all states that render the constraint $e(k) \in \mathcal{B}(\epsilon)$ satisfied irrespective of θ. Then, the set

$$P(r,\epsilon) := \text{MCPI set for (1) w.r.t. } \mathcal{X}_{\mathcal{E}}(r,\epsilon) \times \mathcal{U} \tag{6}$$

is a controlled invariant set inside of which the tracking error constraint $e(k) \in \mathcal{B}(\epsilon)$ is satisfied.

Definition 8. Let $\epsilon \in \mathbb{R}_+$ be given. A reference value $r \in \mathcal{Y}$ is called *ϵ-achievable* for the system (1) and with respect to the constraints $\mathcal{X} \times \mathcal{U}$ if $P(r,\epsilon) \neq \emptyset$.

So far, it was assumed that both the reference and the error bound were known and fixed. The set of all references which are ϵ-achievable for (1) in the sense of Definition 8 is defined as

$$\mathcal{R}(\epsilon) := \{r \in \mathcal{Y} \mid P(r,\epsilon) \neq \emptyset\}. \tag{7}$$

To characterize this set, consider the extended system

$$\begin{bmatrix} x(k+1) \\ r(k+1) \end{bmatrix} = \begin{bmatrix} A(\theta(k)) & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} x(k) \\ r(k) \end{bmatrix} + \begin{bmatrix} B(\theta(k)) \\ 0 \end{bmatrix} u(k)$$
$$e(k) = [C(\theta(k)) \;\; -I] \begin{bmatrix} x(k) \\ r(k) \end{bmatrix} \qquad (8)$$

and the associated $(n_\mathrm{x} + n_\mathrm{y})$-dimensional constraint set

$$\tilde{\mathcal{X}}_\mathcal{E}(\epsilon) := (\mathcal{X} \times \mathcal{Y}) \cap \tilde{\mathcal{E}}(\epsilon)$$

with

$$\tilde{\mathcal{E}}(\epsilon) := \left\{ \begin{bmatrix} x \\ r \end{bmatrix} \in \mathbb{R}^{n_\mathrm{x}+n_\mathrm{y}} \;\middle|\; \forall \theta \in \Theta : \; [C(\theta) \;\; -I] \begin{bmatrix} x \\ r \end{bmatrix} \in \mathcal{B}(\epsilon) \right\}$$

containing all pairs (x, r) satisfying the error bound for all $\theta \in \Theta$. Note that in (8), if the "initial" reference is $r(0) = r_0$, then $r(k) = r_0$ for all $k \geq 0$. This system can be viewed as a special case of the setup from (Di Cairano and Borrelli, 2016), where the reference evolves according to more general non-autonomous constrained LTI dynamics. The invariant set

$$\tilde{P}(\epsilon) := \text{MCPI set for (8) w.r.t. } \tilde{\mathcal{X}}_\mathcal{E}(\epsilon) \times \mathcal{U} \qquad (9)$$

can now be computed. It is a subset of an extended state/reference space of dimension $n_\mathrm{x} + n_\mathrm{y}$. The set $\mathcal{R}(\epsilon)$ is found by projecting $\tilde{P}(\epsilon)$ onto the reference space, i.e.,

$$\mathcal{R}(\epsilon) = \left\{ r \in \mathcal{Y} \;\middle|\; \exists x \in \mathcal{X} : \; \begin{bmatrix} x \\ r \end{bmatrix} \in \tilde{P}(\epsilon) \right\}.$$

Similarly, given $\tilde{P}(\epsilon)$ and a $r \in \mathcal{R}(\epsilon)$, the corresponding set $P(r, \epsilon)$ from (6) can be recovered by projection as

$$P(r, \epsilon) = \left\{ x \in \mathcal{X} \;\middle|\; \begin{bmatrix} x \\ r \end{bmatrix} \in \tilde{P}(\epsilon) \right\}.$$

The projection above can be computed efficiently, because it only considers one fixed reference r. If $r \neq \mathcal{R}(\epsilon)$, i.e., the reference is not ϵ-achievable, the result will be $P(r, \epsilon) = \emptyset$.

Finally, for a given reference r, it can be useful to know the minimal ϵ for which $P(r, \epsilon) \neq \emptyset$. This minimal achievable error bound is defined as

$$\epsilon^\star(r) := \inf \{ \epsilon \geq 0 \mid P(r, \epsilon) \neq \emptyset \} \qquad (10)$$

and the corresponding invariant set is denoted by

$$P^\star(r) := P(r, \epsilon^\star(r)). \qquad (11)$$

In principle, the optimal value $\epsilon^\star(r)$ in (10) can be computed by a bisection search. Such a procedure can be carried out efficiently for low-order systems.

3.2 Time-varying terminal constraint

When a set $P(r, \epsilon)$ has been obtained using the procedures from the previous subsection, it can be used as a terminal set in an MPC algorithm to achieve recursive feasibility. To obtain not only feasibility but also finite-time convergence to $P(r, \epsilon)$, we propose the time-varying terminal constraint

$$\forall i \in \mathbb{N}_{[\max\{0, N-k\}, N]} : \; X_{i|k} \subseteq P(r, \epsilon) \qquad (12)$$

which at the initial time $k = 0$ constrains the state of the system to be inside $P(r, \epsilon)$ after N steps. At time $k+1$, the state is required to reach $P(r, \epsilon)$ in $N-1$ steps: the pattern continues until it is finally required that the initial state $x_{0|k+N} = x(k+N)$ is inside $P(r, \epsilon)$. Given this terminal constraint, a sufficient condition for recursive feasibility is to choose the set S in (3) equal to $P(r, \epsilon)$. Then, we obtain the set of decision variables

$$\begin{aligned} \mathcal{D}_N(k, x, \theta) = \big\{ \mathbf{d} \in \mathbb{D} \;\big|\; & X_{0|k} = \{x\}, \\ & \forall i \in \mathbb{N}_{[\max\{0, N-k\}, N]} : \; X_{i|k} \subseteq P(r, \epsilon), \\ & \forall i \in \mathbb{N}_{[0, N]} : \; \big(X_{i|k}, \Pi_{i|k}\big) \text{ satisfies Def. 5} \big\} \end{aligned} \qquad (13)$$

based on which we can establish recursive feasibility and finite-time convergence to the target set.

Proposition 9. In Equation (3), let $S = P(r, \epsilon)$ and suppose that (5) is feasible at the initial time k, i.e., $\mathcal{D}\big(k, x_{0|k}, \theta_{0|k}\big) \neq \emptyset$. Apply the input $u(k) = \Pi^\star_{0|k}\big(x_{0|k}, \theta_{0|k}\big)$ where $\Pi^\star_{0|k}(\cdot, \cdot)$ follows from the optimal solution of (5). Then, it is guaranteed that $\mathcal{D}\big(k+1, x_{1|k}, \theta_{1|k}\big) \neq \emptyset$ and that the state of the system enters $P(r, \epsilon)$ in at most N steps.

Proof. The solution $\mathbf{d}_k \in \mathcal{D}\big(k, x_{0|k}, \theta_{0|k}\big)$ contains all the parameters $p_{i|k}$, $i \in \mathbb{N}_{[0,N]}$ parameterizing the tube $\mathbf{T}_k = \big(\{X_{i|k}\}_0^N, \{\Pi_{i|k}\}_0^{N-1}\big)$. We can assume without loss of generality that $k = 0$, so the terminal constraint (12) becomes $X_{N|k} \subseteq P(r, \epsilon)$. Then, at time $k+1$, at least one tube satisfying the updated constraint $\forall i \in \mathbb{N}_{[N-1,N]} : \; X_{i|k+1} \subseteq P(r, \epsilon)$ exists. It is explicitly given as $\mathbf{T}^\circ_{k+1} = \big(\{\{x_{0|k+1}\}, X_{2|k}, \ldots, X_{N-1|k}, P(r, \epsilon), P(r, \epsilon)\}, \{\Pi_{1|k}, \Pi_{2|k}, \ldots, \Pi_{N-2|k}, \Pi_\mathrm{p}\}\big)$ where $\Pi_\mathrm{p}(\cdot, \cdot)$ is the vertex controller induced by the set $P(r, \epsilon)$. The homothetic- and vertex control parameterizations (3)-(4) guarantee existence of $p_{i|k+1}$ such that $\forall i \in \mathbb{N}_{[0,N]} : \; \big(X_{i|k+1}, \Pi_{i|k+1}\big) = \bar{P}\big(p_{i|k+1}\big)$ (Hanema et al., 2016, Lemma 2). Thus, $\mathbf{T}_{k+1}$ is a valid parameterized tube proving the existence of at least one feasible solution $\mathbf{d}^\circ_{k+1} \in \mathcal{D}\big(k+1, x_{0|k+1}, \theta_{0|k+1}\big)$. By induction, it can be concluded that the problem remains feasible for all $k+i$, $i \in \mathbb{N}_{[2,\infty)}$. Further, for $i \in \mathbb{N}_{[N,\infty)}$, it holds that $x_{0|k+i} \in P(r, \epsilon)$ by construction of (12), proving finite-time convergence to $P(r, \epsilon)$ in at most N steps. □

The idea of Proposition 9 is similar to that of the "decreasing horizon tube controller" from (Langson et al., 2004, Proposition 7), where finite-time convergence to a robustly invariant set is established for an LTI system subject to additive disturbances. The main difference is that we vary the time at which the terminal constraint becomes active instead of the horizon length N itself, preserving degrees of freedom to optimize local performance once $P(r, \epsilon)$ has been reached. Finally, observe that the proof of Proposition 9 does not require $e_{0|k} \in \mathcal{B}(\epsilon)$, where $e_{0|k} = x_{0|k} - r$.

3.3 Cost function for tracking

The third and final component in our setup is the cost function $J_N(\cdot, \cdot)$. It is chosen as a standard, possibly time-varying, finite-horizon cost, i.e.,

$$J_N(k, \mathbf{d}) = \sum_{i=0}^{N-1} \ell\big(k, X_{i|k}, \Pi_{i|k}\big) \qquad (14)$$

where $\ell(\cdot, \cdot, \cdot)$ is the stage cost. The finite-time convergence established in Proposition 9 is independent of the cost used in (5). Therefore, a terminal cost is not required and there is some freedom in the stage cost design. The stage cost

$$\ell(k, X, \Pi) = \max_{(x,\theta)\in X\times\Theta} \left\| Q\left(x - \bar{x}(r, \theta_{0|k})\right)\right\| + \left\| R\left(\Pi(x,\theta) - \bar{u}(r, \theta_{0|k})\right)\right\| \quad (15)$$

will be used. Here,

$$(\bar{x}(r,\theta), \bar{u}(r,\theta)) = \arg\min_{(\bar{x},\bar{u})} \left\| \begin{bmatrix}\bar{x}^\top & \bar{u}^\top\end{bmatrix}\right\|_2^2 \quad \text{s.t.} \begin{bmatrix} A(\theta) - I & B(\theta) \\ C(\theta) & 0 \end{bmatrix} \begin{bmatrix}\bar{x}\\ \bar{u}\end{bmatrix} = \begin{bmatrix}0\\ r\end{bmatrix}$$

which always has a solution due to Assumption 2.(iv). In the above equation, we selected a least-norm cost function, preferring solutions with small input energy. However, if $n_y = n_u$, there exists only one unique solution regardless of the minimization objective. The main reasoning behind selecting the stage cost (15) is that it brings the output close to the reference if θ is slowly varying, and possibly even yields offset-free tracking if θ stops varying (i.e., if $\exists k_\star \in \mathbb{N}$ such that $\forall k \geq k_\star :\ \theta(k) = \bar{\theta}$).

Remark 10. Offset-free tracking under constant θ is not guaranteed in the considered control solution, because this objective is not necessarily compatible with the bounded-error requirement that $e(k) \in \mathcal{B}(\epsilon)$ for all k. It can happen that there exists $\theta \in \Theta$ for which $\bar{x}(r,\theta) \notin P(r,\epsilon)$: an example of such a case is given in the next section.

4. NUMERICAL EXPERIMENTS

It was shown in a previous work (Hanema et al., 2016) that (5), under some conditions, can be written as a single *linear program* (LP) whose size grows linearly with the prediction horizon N. This remains true for the design choices made in this paper. In particular, observe that minimization of (14) with the infinity-norm based stage cost (15) is equivalent to the minimization of a linear function by introducing slack variables. Furthermore, since all considered sets are polytopes, all constraints in (13) are linear. Following the implementation of (Hanema et al., 2016), the dominant term specifying the number of constraints necessary to check the set inclusion conditions of Definition 4 is $q_s r_s q$, where q_s is the number of vertices of $P(r,\epsilon)$, r_s is the number of its hyperplanes (which can be smaller or larger than q_s), and q is the number of vertices of Θ. Hence, the computational complexity of the approach is mainly determined by the complexity of the invariant sets $P(r,\epsilon)$.

We now provide two numerical examples demonstrating the tracking LPV algorithm developed in the previous section.

4.1 Example 1: minimal ϵ for fixed r

In this example, a fixed constant reference r is given and we want to track it with the minimal achievable error bound $\epsilon^\star$ defined in (10). The system used is of the form (1) with

$$A_0 = \begin{bmatrix} 0.95 & 1 \\ 0 & -0.59\end{bmatrix},\ A_1 = \begin{bmatrix} 0 & 0.5 \\ 0 & 0\end{bmatrix},\ A_2 = \begin{bmatrix} 0 & 0 \\ 0.2 & 0\end{bmatrix}$$
$$B = [1\ 0.5]^\top,$$
$$C_0 = [0.8\ -0.6],\ C_1 = [0\ -0.03],\ C_2 = [0.04\ 0]$$
$$\Theta = \left\{\theta \in \mathbb{R}^2 \mid \|\theta\| \leq 1\right\},\ \mathcal{U} = \{u \in \mathbb{R} \mid -1 \leq u \leq 1\},$$
$$\mathcal{X} = \left\{x \in \mathbb{R}^2 \mid -6 \leq x_1 \leq 4,\ -4 \leq x_2 \leq 6\right\},$$

The constant reference to be tracked is $r = -0.85$. Using the procedure from Section 3, it was found that $\epsilon^\star = 0.37$.

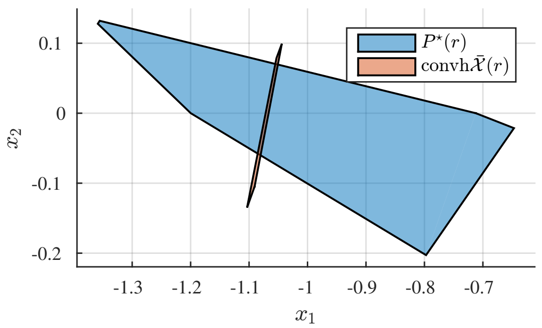

Fig. 1. The sets $P^\star(r)$ and $\text{convh}\bar{\mathcal{X}}(r)$ in Example 1.

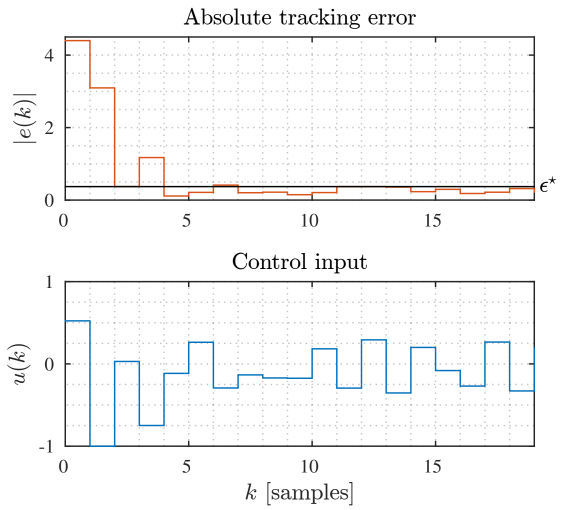

Fig. 2. Simulation result for Example 1.

The corresponding set $P^\star(r)$ is shown in Figure 1. In light of Remark 10, the convex hull of the set of all steady states $\bar{\mathcal{X}}(r) = \left\{\bar{x}(r,\theta) \mid \theta \in \Theta\right\}$ is also depicted. It can be clearly seen that there exist $\theta \in \Theta$ for which offset-free tracking of r is, by definition, impossible to achieve without violating the guarantee that $e(k) \in \mathcal{B}(\epsilon^\star)$ for all time given all possible trajectories of θ. A tube-based tracking MPC was implemented according to the construction of Section 3. This simulation used $N = 7$, $Q = I$ and $R = 1$. A simulation result for the initial state $x_{0|k} = \left[-4\ 3\right]^\top$ is shown in Figure 2. The tracking error converges to $\mathcal{B}(\epsilon)$ in less than N steps as predicted by Proposition 9. The scheduling signal used in this simulation corresponds to a "harsh" scenario, where at each time time instant k the value $\theta(k)$ was a randomly selected vertex of the set Θ.

4.2 Example 2: all ϵ-achievable references

In this second example, a desired error bound ϵ is specified. We are interested in obtaining the set $\mathcal{R}(\epsilon)$ of all ϵ-achievable references according to (7). The system used in this example is mostly the same as in Example 1. To make the computation of $\mathcal{R}(\epsilon)$ more interesting, a second input and output are added by setting

$$B = \begin{bmatrix} 1 & 0.7 \\ 0.5 & -1\end{bmatrix},\ C_0 = \begin{bmatrix} 1.2 & 1.0 \\ 0.7 & 0\end{bmatrix},$$
$$C_1 = \begin{bmatrix} 0 & -0.03 \\ 0.08 & 0\end{bmatrix}, C_2 = \begin{bmatrix} 0.04 & 0 \\ 0 & -0.06\end{bmatrix},$$

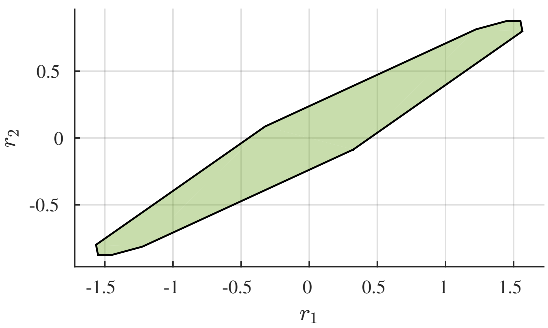

Fig. 3. The set $\mathcal{R}(\epsilon)$ obtained in Example 2.

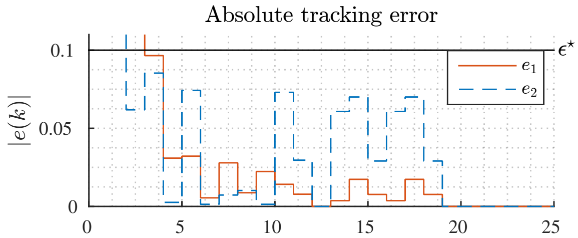

Fig. 4. Tracking error of the MPC in Example 2.

$$\mathcal{U} = \left\{ u \in \mathbb{R}^2 \mid -1 \leq u_1 \leq 1,\ -0.4 \leq u_2 \leq 0.4 \right\}.$$

A desired tracking error bound of $\epsilon = 0.1$ is specified. The corresponding set $\mathcal{R}(\epsilon) \subset \mathbb{R}^{n_y}$ was calculated according to the procedures from Section 3 and the result is shown in Figure 3. Based on this analysis we choose to track the constant reference $r = [\,0.5\ 0.3\,]^\top$, which is in the interior of $\mathcal{R}(\epsilon)$. A corresponding simulation result, for the initial state $x_{0|k} = [\,2\ 2\,]^\top$ is displayed in Figure 4. The prediction horizon $N = 7$ and tuning settings $Q = I$ and $R = I$ were the same as in the previous experiment. The scheduling signal consisted of random switching between the vertices of Θ until the 18th sample, after which $\theta(k)$ remained equal to its last value. In Figure 4, it can be observed that the tracking error goes to zero: thus, in this special case the controller achieves offset-free tracking. However, there also exist references $r \in \mathcal{R}(\epsilon)$ for which this is not possible (see Remark 10). An example of such a case was already considered in the previous experiment, where in Figure 1, it was shown that $\operatorname{convh}\bar{\mathcal{X}}(r) \nsubseteq P(r, \epsilon)$.

5. CONCLUDING REMARKS

A tube-based MPC algorithm to track constant references for LPV systems with a guaranteed error bound was presented. The results of this paper are the first step towards more sophisticated reference-tracking LPV predictive controllers. In particular, we aim to develop an approach which combines a guaranteed ϵ-error bound with the ability to always achieve asymptotically offset-free tracking if $\theta(k)$ stops varying. An issue with the current method is that the domain of attraction of the controller can be small if ϵ is chosen small. Hence, we want to construct an extended terminal set around $P(r, \epsilon)$ and use an appropriate terminal cost to ensure convergence, possibly without the time-varying terminal constraint. Finally, the approach could be extended to a larger class of references, e.g., piecewise-constant signals or those generated by *reference generators* as in (Di Cairano and Borrelli, 2016).

REFERENCES

Alvarado, I., Limón, D., Alamo, T., Fiacchini, M., and Camacho, E.F. (2007). Robust tube based MPC for tracking of piece-wise constant references. In *Proc. of the 46th IEEE Conference on Decision and Control*, 1820–1825.

Betti, G., Farina, M., and Scattolini, R. (2013). A Robust MPC Algorithm for Offset-Free Tracking of Constant Reference Signals. *IEEE Transactions on Automatic Control*, 58, 2394–2400.

Blanchini, F. and Miani, S. (2008). *Set-Theoretic Methods in Control*. Birkhäuser.

Blanchini, F., Miani, S., and Savorgnan, C. (2007). Stability results for linear parameter varying and switching systems. *Automatica*, 43, 1817–1823.

Casavola, A., Famularo, D., and Franzè, G. (2002). A Feedback Min-Max MPC Algorithm for LPV Systems Subject to Bounded Rates of Change of Parameters. *IEEE Transactions on Automatic Control*, 47, 1147–1153.

Di Cairano, S. and Borrelli, F. (2016). Reference Tracking With Guaranteed Error Bound for Constrained Linear Systems. *IEEE Transactions on Automatic Control*, 61, 2245–2250.

Falugi, P. and Mayne, D.Q. (2013). Model predictive control for tracking random references. In *Proc. of the 2013 European Control Conference*, 518–523.

Fleming, J., Kouvaritakis, B., and Cannon, M. (2015). Robust Tube MPC for Linear Systems With Multiplicative Uncertainty. *IEEE Transactions on Automatic Control*, 60, 1087–1092.

Hanema, J., Tóth, R., and Lazar, M. (2016). Tube-based anticipative model predictive control for linear parameter-varying systems. In *Proc. of the 55th IEEE Conference on Decision and Control*, 1458–1463.

Hoffmann, C. and Werner, H. (2014). A Survey of Linear Parameter-Varying Control Applications Validated by Experiments or High-Fidelity Simulations. *IEEE Transactions on Control Systems Technology*, 23, 416–433.

Langson, W., Chryssochoos, I., Raković, S.V., and Mayne, D.Q. (2004). Robust model predictive control using tubes. *Automatica*, 40, 125–133.

Lu, Y. and Arkun, Y. (2000). Quasi-Min-Max MPC algorithms for LPV systems. *Automatica*, 36, 527–540.

Mohammadpour, J. and Scherer, C.W. (2012). *Control of linear parameter varying systems with applications*. Springer.

Muñoz-Carpintero, D., Cannon, M., and Kouvaritakis, B. (2015). Robust MPC strategy with optimized polytopic dynamics for linear systems with additive and multiplicative uncertainty. *Systems & Control Letters*, 81, 34–41.

Pannocchia, G. (2004). Robust model predictive control with guaranteed setpoint tracking. *Journal of Process Control*, 14, 927–937.

Pannocchia, G., Gabiccini, M., and Artoni, A. (2015). Offset-free MPC explained: novelties, subtleties, and applications. In *Proc. of the 5th IFAC Conference on Nonlinear Model Predictive Control*, 342–351.

Wang, Y.J. and Rawlings, J.B. (2004). A new robust model predictive control method I: Theory and computation. *Journal of Process Control*, 14, 231–247.