Proceedings of the
47th IEEE Conference on Decision and Control
Cancun, Mexico, Dec. 9-11, 2008

Explicit Model Predictive Control for Linear Parameter-Varying Systems

Thomas Besselmann, Johan Löfberg and Manfred Morari

Abstract— **In this paper we demonstrate how one can reformulate the MPC problem for LPV systems to a series of mpLPs by a closed-loop minimax MPC algorithm based on dynamic programming. A relaxation technique is employed to reformulate constraints which are polynomial in the scheduling parameters to parameter-independent constraints. The algorithm allows the computation of explicit control laws for linear parameter-varying systems and enables the controller to exploit information about the scheduling parameter. This improves the control performance compared to a standard robust approach where no uncertainty knowledge is used, while keeping the benefits of fast online computations. The off-line computational burden is similar to what is required for computing explicit control laws for uncertain or nominal LTI systems. The proposed control strategy is applied to an example to compare the complexity of the resulting explicit control law to the robust controller.**

I. Introduction

The study of Linear Parameter-Varying (LPV) systems is motivated by their use in gain-scheduling control techniques, [1], [2]. Classical gain-scheduling approaches work with an interpolation of the controller gains among a family of LTI controllers, which are based on linearized models of the system. While those techniques work surprisingly well in practice, it is hard to give precise stability/performance statements taking changes in the system dynamics into account. LPV systems account for changes in the system dynamics by parameter-varying system matrices. Contrary to systems with parametric uncertainties, the current values of the scheduling parameters are known. The parameters lie in a bounded set, such that an LPV system describes a *family* of linear systems. The LPV framework constitutes a useful theoretical foundation and allows statements on stability and performance which take variations of the scheduling parameter directly into account, [3].

Linear parameter-varying systems were also considered in the Model Predictive Control (MPC) community, and various approaches were developed for discrete-time LPV systems. These include quasi-min-max MPC, [4], which demands the solution of semi-definite programs, interpolation-based MPC, [5], which relies on the existence of robustly stabilizing controllers, or gain-scheduling MPC, [6], where non-convex optimization is avoided by using a modified cost objective.

The introduction of multi-parametric programming into the field of model predictive control around the millennium now allows the computation of explicit solutions for the optimal control problem of constrained linear and piecewise affine systems, [7], [8], [9], [10], [11]. Instead of solving an optimization problem at each sampling instance, the optimal input is obtained from a look-up table, which significantly reduces the online computational effort. In [12], [13], the computation of explicit control laws was extended to linear discrete-time systems with parametric uncertainty, i.e., when the parameter is bounded, but *unknown*. The computation of explicit control laws for LPV systems - when the scheduling parameter is *known* - was presented recently for the case of constant input matrices, [14], [15]. In order to tackle the whole class of LPV systems, a more demanding procedure is needed due to the occurrence of polynomials in the scheduling parameter.

In the following we are proposing an MPC scheme for LPV systems which results in a series of multi-parametric linear programs (mp-LPs), i.e., multi-parametric programming can be employed to pre-compute the explicit solution offline.

The paper is structured as follows: In Section II, the considered problem is stated. The main results are presented in Section III, followed by a brief discussion on the relaxation technique in Section IV. The aspect of stability is treated in Section V, before an numerical example illustrates the application of the algorithm in Section VI. Finally conclusions are drawn.

A. Notation

The set of non-negative real numbers is denoted by $\mathbb{R}_+$. The positive orthant in the n-dimensional Euclidean space is denoted by $\mathbb{R}^n_+$. A *polyhedron* is a set described by the intersection of finitely many half-spaces. A *polytope* is a closed and bounded polyhedron. An upper index in brackets denotes the element of a vector or the row in a matrix.

II. Problem Statement

We consider linear discrete-time LPV systems with a parameter-varying state transition and parameter-varying input matrix

$$x_{k+1} = A(\theta_k)x_k + B(\theta_k)u_k\,. \tag{1}$$

J. Löfberg is with the Department of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden, WWW: http://www.control.isy.liu.se, Email: johanl@isy.liu.se

Th. Besselmann and M. Morari are with the Automatic Control Laboratory, ETH Zurich, Switzerland, WWW: http://control.ee.ethz.ch, Email: besselmann |morari @ control.ee.ethz.ch

The variables $x_k \in \mathbb{R}^{n_x}$, $u_k \in \mathbb{R}^{n_u}$ and $\theta_k \in \mathbb{R}_+^{n_\theta}$ denote the state, control input and time-varying scheduling parameter, respectively. Furthermore, the system is constrained, $x_k \in \mathbb{X}$ and $u_k \in \mathbb{U}$. The constraint sets $\mathbb{X}$ and $\mathbb{U}$ are assumed to be polytopes including the origin in its interior,

$$\mathbb{X} = \{x : E_x x \le f_x\}, \tag{2a}$$
$$\mathbb{U} = \{u : E_u u \le f_u\}. \tag{2b}$$

Remark 1: We restrict ourselves to separate constraints on the state and inputs in (2) solely for ease of notation. It is straight-forward to modify the presented algorithm in this paper to the case of mixed constraints, i.e. $E_x x + E_u u \le f_{xu}$.

The scheduling parameter vector $\theta_k = [\theta_k^{[1]}, \dots, \theta_k^{[n_\theta]}]^T$ is measured online. Future values are however only known to be the barycentric coordinates of a standard n_θ-simplex Θ in the parameter space,

$$\Theta := \{\theta_k \in \mathbb{R}_+^{n_\theta} : \sum_{j=1}^{n_\theta} \theta_k^{[j]} = 1\}. \tag{3}$$

The parameter-varying matrices $A(\theta_k)$ and $B(\theta_k)$ are known to lie in polytopes,

$$A(\theta_k) = \sum_{j=1}^{n_\theta} A_j \theta_k^{[j]}, \quad B(\theta_k) = \sum_{j=1}^{n_\theta} B_j \theta_k^{[j]}, \tag{4}$$

where A_j and B_j denote the j-th vertices of the polytopes. This polytopic description is a common assumption in the LPV framework, see e.g. [2]. For the control problem to make sense, it is assumed that the system (1) is controllable (and observable) for all $\theta_k \in \Theta$, see [16], [17]. For this class of systems we want to compute an explicit state-feedback control law

$$u_k = \mu_k(x_k, \theta_k), \tag{5}$$

which makes use of knowledge of the current scheduling parameter θ_k. In order to compute this control law (5) within a Model Predictive Control scheme, a cost function is to be minimized. The control objective is to stabilize the LPV system (1) to the origin. According to standard MPC, our cost function is defined as

$$J = \|Px_{k+N}\|_p + \sum_{i=0}^{N-1} \|Qx_{k+i}\|_p + \|Ru_{k+i}\|_p, \tag{6}$$

where p denotes a piecewise linear norm, either the 1-norm or the ∞-norm. Piecewise linear norms[1] enable a parametric solution to the stated problem using dynamic programming. For the minimization of the cost function (6) we have to consider the current as well as the unknown future scheduling parameter values, as the state trajectories are parameter-dependent.

[1]Quadratic cost functions are not possible since our procedure relies on epigraph reformulations, which would render the original problem a multiparametric quadratically constrained quadratic program, for which no efficient solution techniques are available.

III. Main Results

Before starting the actual computation of the control law (5), the polynomial dependency of the control law μ on the scheduling parameter has to be decided. This dependency can chosen to be affine

$$\mu_k(x_k, \theta_k) = \sum_{j=1}^{n_\theta} \theta_k^{[j]} \mu_k^j(x_k), \tag{7}$$

but note that in principle any polynomial in the parameter θ_k is possible with our proposed method. In the case of the affine parametrization (7), the function $\mu_k^j(x_k)$ corresponds to the control law in the j-th vertex of the parameter simplex (3). In order to simplify notation, we introduce the basis $U_k := \{\mu_k^1, \mu_k^2, \dots, \mu_k^{n_\mu}\}$.

In a closed-loop MPC approach, one would assume that the future control law μ_{k+i} is calculated optimally over the horizon $N-i$ not until x_{k+i} and θ_{k+i} are available. But as the future values of the scheduling parameters are unknown, all possible cases must be considered in order to accommodate for the worst-case scenario. This way constraint satisfaction is assured and the actual cost function will be less or equal to the computed one, no matter how the scheduling parameters evolve. The optimization problem to solve in closed-loop minimax MPC is thus

$$\mu_k(x_k, \theta_k) = \arg\min_{U_k} \max_{\theta_k} \cdots \min_{U_{k+N-1}} \max_{\theta_{k+N-1}} J \tag{8}$$

Here we propose a *dynamic programming* (DP) procedure to solve (8) by iterating backwards in time. For more details on dynamic programming, see [18]. We start at the prediction horizon N with the initial cost function

$$J_N^*(x_{k+N}) = \|Px_{k+N}\|_p. \tag{9}$$

Then at each iteration we use

$$x_{k+i+1} = A(\theta_{k+i})x_{k+i} + B(\theta_{k+i})\mu_{k+i}(x_{k+i}, \theta_{k+i}) \tag{10}$$

to substitute x_{k+i+1} in $J_{i+1}^*(x_{k+i+1})$. As θ_{k+i} is unknown at time instance k, we consider the worst case, which leads to

$$\begin{aligned} J_i^*(x_{k+i}) &= \min_{U_{k+i}} \max_{\theta_{k+i}} \|Qx_{k+i}\|_p + \|Ru_{k+i}\|_p \\ &+ J_{i+1}^*(x_{k+i+1}). \end{aligned} \tag{11}$$

In order to determine the worst-case parameters of (11), we first apply an epigraph reformulation to the optimization problem in order to transfer the parameter dependence to the constraints. This leads to the following semi-infinite optimization problem

$$J_i^*(x_{k+i}) = \min_{U_{k+i}} t \tag{12a}$$

s.t. $\forall \theta_{k+i} \in \Theta$:

$$\|Qx_{k+i}\|_p + \|Ru_{k+i}(\theta_{k+1})\|_p + J_{i+1}^*(A(\theta_{k+i})x_{k+i} + B(\theta_{k+i})\mu_{k+i}(x_{k+i}, \theta_{k+i})) \le t, \tag{12b}$$

$$x_{k+i} \in \mathbb{X}, \qquad u_{k+i} \in \mathbb{U}. \tag{12c}$$

Remark 2: Depending on the structure of (10), the assurance of constraint satisfaction differs. If the control law (5) is chosen to be parameter-independent, the constraints in (12) are convex in the scheduling parameters and the maximum is attained at one of the vertices of the parameter simplex Θ. However, the resulting control law does not take the current scheduling parameters into account, [12], [19], [13]. For a constant input matrix B, it is reasonable to assume a polytopic input parametrization. The resulting constraints depend affinely on the scheduling parameter, which again allows for vertex enumeration. This case was tackled in detail in [14].

In the more general case of a non-constant input matrix $B(\theta_k)$ and a polynomially parameterized input $u_k(\theta_k)$, the constraints are polynomial in the scheduling parameters and a vertex enumeration is *not* sufficient to ensure constraint satisfaction over the whole simplex. However, the constraint satisfaction of the semi-infinite optimization problem (12) can be ensured, conservatively, over the whole parameter simplex by making use of Pólya's theorem:

Theorem 1: **Pólya's theorem.** If a homogeneous polynomial $p(\theta)$ is positive on the simplex Θ, all the coefficients of $p_{N_p}(\theta) = p(\theta)\cdot(\sum_{j=1}^{n_\theta}\theta^{[j]})^{N_p}$ are positive for a sufficiently large Pólya degree N_p.

Proof: See [20], [21]. ■

We will make use of the more obvious reverse of Pólya's theorem[2], i.e., positive coefficients of the extended polynomial mean positivity over the whole simplex.

Example 1: Consider the polynomial $p(\theta) = a(\theta^{[1]})^2 + b\theta^{[1]}\theta^{[2]} + c(\theta^{[2]})^2$. A sufficient condition for positivity over the standard simplex Θ is the positivity of the coefficients $c_0 = \{a, b, c\}$. By multiplying with $\sum_{j=1}^{n_\theta}\theta^{[j]}$, we obtain $p_1(\theta) = a(\theta^{[1]})^3 + (a+b)(\theta^{[1]})^2\theta^{[2]} + (b+c)\theta^{[1]}(\theta^{[2]})^2 + c(\theta^{[2]})^3$ and the less conservative condition of positive coefficients $c_1 = \{a, a+b, b+c, c\}$. Pólya's contribution was to show that by repeated multiplication with $\sum_{j=1}^{n_\theta}\theta^{[j]}$, the condition of positive coefficients indeed converges to the exact necessary and sufficient condition for positivity of the polynomial over the standard simplex. Fig. 1 shows the resulting conditions on the coefficients of $p(\theta)$ for different Pólya degrees in the case $a = c$.

The following design procedure describes the relaxation of the parameter-dependent constraints of (12) into constraints which are piecewise affine in the state and inputs and independent of the scheduling parameter:

1) Reformulate constraints which are polynomial in the scheduling parameter into a positivity constraint of a polynomial $p(\theta)$.
2) Homogenize the polynomial $p(\theta)$ by multiplying single monomials with $\sum_{j=1}^{n_\theta}\theta^{[j]}(=1)$ until all monomials have the same degree.
3) Set the Pólya degree N_p, and compute the coefficients c_{N_p} of the extended polynomial $p_{N_p}(\theta) = p(\theta)\cdot(\sum_{j=1}^{n_\theta}\theta^{[j]})^{N_p}$. In this step some conservatism may be introduced depending on the selection of N_p. By increasing the polynomial degree N_p, the relaxations become tighter until the exact problem is considered.

[2]The presented usage of Pólya's theorem is implemented in YALMIP as one of the so called filters in the robust optimization framework, [22].

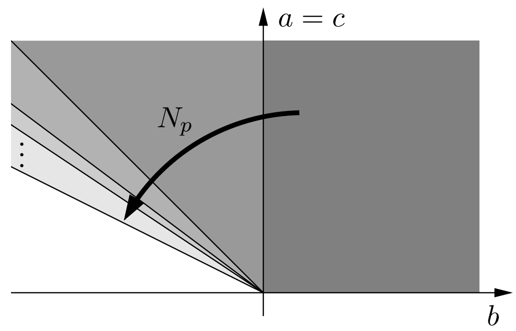

Fig. 1. Abating conditions on the coefficients of $p(\theta)$ for increasing Pólya degrees $N_p = 0, 1, 3, 5, ..., \infty$.

If all coefficients c_{N_p} are non-negative, so is the polynomial $p(\theta)$. Hence the semi-infinite optimization problem (12) can be transformed into the following multi-parametric linear program:

$$J_i^*(x_{k+i}) = \min_{U_{k+i}} t \qquad (13a)$$

s.t.

$$c_{N_p}(x_{k+i}, U_{k+i}, t) \geq 0\,, \qquad x_{k+i} \in \mathbb{X} \qquad (13b)$$

Note that the coefficients of the extended polynomial lie in the cone which is spanned by the coefficients of the polynomial constraints in (12), and the piecewise affine dependence of the coefficients on the state is preserved. By using piecewise linear norms instead of quadratic norms, the cost functions J_i^* are piecewise linear functions of the state x_{k+i}, such that in every iteration the optimization problem (12) can be formulated as the multi-parametric linear program (13) and solved parametrically with respect to x_{k+i}. Contrary to the closed-loop minimax MPC approach for uncertain systems, the future inputs are functions of the future scheduling parameters.

The final step of the DP procedure differs from the preceding steps, since knowledge of the current scheduling parameter values can be exploited to improve control performance. We make use of the *uncontrolled successor state* (USS),

$$z_k = (\sum_{j=1}^{n_\theta} A_j\theta_k^{[j]})x_k, \qquad (14)$$

which was first introduced in [14] and in generalized form constitutes a cornerstone of [15]. By parameterizing the parametric problem not in the measured state x_k, but in the USS, the parameter dependence of $A(\theta_k)$ can directly be taken into account. In lack of an equivalent scheme for the input, we minimize the worst-case gain-scheduled cost, i.e. solve the semi-infinite optimization problem

$$J^*(z_k) = \min_{U_k}\ t + s \qquad (15a)$$

s.t. $\forall \theta_k \in \Theta$:

$$\|R\mu_k(z_k, \theta_{k+1})\|_p \; + J_1^*(z_k + B(\theta_k)\mu_k(z_k, \theta_k)) \leq t\,, \tag{15b}$$

$$u_k \in \mathbb{U}\,, \tag{15c}$$

$$\epsilon \sum_{j=1}^{n_\theta} \|Q(z_k + B(\theta_k^{[j]})\mu_k(z_k, \theta_k^{[j]}))\|_p \leq s\,, \tag{15d}$$

employing an additional epigraph variable s. For many problems, the vertex solutions are not unique, which can lead to irregular control laws when solving the parametric problem, and therefore a small regularization weight, $0 < \epsilon \ll 1$, which penalizes the vertex predictions, is added as an ad-hoc measure. While this virtually does not change the shape of the cost function, it turns out that the actual achieved performance is improved, because the non-uniqueness of the vertex solutions is mitigated. The polynomial dependence on the scheduling parameter can again be treated by employing Pólya's theorem to transform (15) into an mp-LP. The resulting control law $\mu_k(z_k, \theta_k)$ is piecewise affine in the USS, defined over a set of polytopes in the USS-space, and polynomial in the scheduling parameter, such that is can be written as

$$\mu_k(z_k, \theta_k) = F_r(\theta_k) z_k + g_r(\theta_k) \quad \text{if} \quad z_k \in \mathbb{D}_r \tag{16}$$

where

$$\mathbb{D}_r = \{z : E_r z \leq f_r\} \quad r = \{1, \ldots, n_r\}\,. \tag{17}$$

The union of these polytopes is the set of feasible uncontrolled successor states,

$$\mathbb{Z}_f = \{z : E_z z \leq f_z\}, \tag{18}$$

and the set of *admissible* initial states, i.e. which result in a feasible USS for all parameter values, can be determined by

$$\mathbb{X}_f = \{x : \begin{bmatrix} E_z A_1 \\ \vdots \\ E_z A_{n_\theta} \end{bmatrix} x \leq \begin{bmatrix} f_z \\ \vdots \\ f_z \end{bmatrix}\}\,. \tag{19}$$

When applying the computed control law online, in each step the state x_k and the scheduling parameter θ_k are measured and used to compute the uncontrolled successor state z_k, which is then inserted in the parametric solution to obtain the control input u_k.

IV. Choice of Pólya degree

When applying Pólya's theorem to ensure constraint satisfaction over the whole simplex Θ, the question remains how to choose the Pólya degree N_p appropriately. Intuitively, the higher the degree the less conservatism is introduced by the relaxation, but the question remains how large N_p has to be chosen in order to guarantee that no conservatism is introduced? In [21] the authors were able to derive an explicit bound for the Pólya degree N_p.

Theorem 2: Suppose that $p(\theta)$ is a homogeneous polynomial of degree d and positive on the simplex Θ. The maximum of the scaled coefficients of $p(\theta)$ is denoted by L and the minimum of the polynomial over the simplex by λ. If

$$N_p > \frac{d(d-1)}{2}\frac{L}{\lambda} - d\,, \tag{20}$$

then $p(\theta) \cdot (\sum_{j=1}^{n_\theta} \theta_j)^{N_p}$ has positive coefficients.

Unfortunately, when the optimum of the parametric optimization problem (13) is obtained, for some $\theta_k \in \Theta$ constraints are fulfilled with equality, i.e., $\lambda \to 0$ for some constraints. Thus for the application in the optimization problem (12) no general bound for a sufficient large Pólya degree N_p can be derived from (20).

However, as will be shown by an example in Section VI, in many cases the introduced conservatism is not severe, and the use of small Pólya degrees leads to a substantial performance improvement compared to a robust MPC scheme.

V. Stability

Another question concerns the stability of the resulting closed-loop system. Note that the proposed procedure does not guarantee stability a-priori, a classical issue of finite horizon MPC. The stability of the control law (16) can be verified a-posteriori by performing a reachability analysis of the closed-loop system in the space of the uncontrolled successor state:

$$z_{k+1} = A(\theta_{k+1})x_{k+1} \;=\; A(\theta_{k+1})\{z_k + B(\theta_k)\mu_k(z_k, \theta_k)\} \tag{21}$$

towards a target region around the origin. First the stability of this target region under the computed control law is established following the theory described in [23], [24].

Definition 1: A function $\Psi : \mathbb{R}^n \to \mathbb{R}_+$ is said to be a *gauge function* if

(a) $\Psi(z + \hat{z}) \leq \Psi(z) + \Psi(\hat{z}), \quad \forall z, \hat{z} \in \mathbb{R}^n$,
(b) $\Psi(z) \geq 0, \Psi(z) = 0 \Leftrightarrow z = 0$,
(c) $\Psi(\mu z) = \mu\Psi(z), \quad \forall \mu \in \mathbb{R}_+$.

Definition 2: A ball of radius r with respect to $\Psi(z)$ is defined as

$$B_r = \{z : \Psi(z) \leq r\}. \tag{22}$$

A convex and compact set $\mathbb{T}$ containing the origin in its interior can be regarded as the unit ball of a gauge function $\Psi_{\mathbb{T}}(z)$. Moreover, if the set is polyhedral, it can be written in standard form:

$$\mathbb{T} = \{z \in \mathbb{R}^n : E_t z \leq 1\}, \tag{23}$$

and induces the gauge function (also known as the Minkowski functional of $\mathbb{T}$)

$$\Psi_{\mathbb{T}}(z) = \max_i E_t^{[i]} z. \tag{24}$$

Definition 3: A set $\mathbb{T}$ is said to be *λ-contractive* if $\forall z_k \in \mathbb{T}, \forall \theta \in \Theta : z_{k+1} \in \lambda\mathbb{T}$.

Definition 4: Let Ω be the set of all controller regions containing the origin,

$$\Omega = \{r : 0 \leq f_r\}\,. \tag{25}$$

Ω is single-valued if the origin is contained in the interior of a controller region, and multi-valued if the origin lies on the facet of several controller regions.

Assumption 1: $g_r(\theta_k) = 0 \; \forall r \in \Omega$.

Note that this assumption has to be fulfilled in order to hold the state in the origin. Since $\mathbb{X}$ and $\mathbb{U}$ include the origin, it is always possible to fulfill Assumption 1.

Proposition 1: Let $\mathbb{T}$ be a polytope, $\mathbb{T} \subseteq \bigcup_{r \in \Omega} \mathbb{D}_r$ and let Assumption 1 hold. If $\forall r \in \Omega$, the vertices v_r^i of $\mathbb{T} \bigcap \mathbb{D}_r$ are mapped into $\lambda\mathbb{T}$, then $\mu\mathbb{T}$ is λ-contractive $\forall 0 \geq \mu \geq 1$.

Proof: Consider any $\hat{z}_k \in \mu(\mathbb{T} \bigcap \mathbb{D}_r) \Rightarrow \tilde{z}_k = \hat{z}_k/\mu \in (\mathbb{T} \bigcap \mathbb{D}_r) \Leftrightarrow \exists \alpha_r^i \in \mathbb{R}_+, \sum_i \alpha_r^i = 1 : \tilde{z}_k = \sum_i \alpha_i v_r^i \Rightarrow \hat{z}_k = \mu \sum_i \alpha_i v_r^i \Rightarrow \hat{z}_{k+1} \in \mu \sum_i \alpha_i \lambda \mathbb{T} = \mu\lambda\mathbb{T}$. ■

Proposition 1, together with the properties of the induced gauge function $\Psi(z)$ suffices to establish Lyapunov stability inside $\mathbb{T}$.

A reachability analysis can be performed to check which states are mapped into $\mathbb{T}$ under the computed control law.

$\mathbb{T}_0 := \mathbb{T}\,,$

$\mathbb{T}_{k+1} := \{z : A(\theta^+)\{z + B(\theta)\mu(z,\theta)\} \in \mathbb{T}_k \forall \theta, \theta^+ \in \Theta\}\,.$

The iteration is terminated, when for a k, $\mathbb{T}_k$ covers the complete feasible space or when the stability region $\mathbb{T}_k$ converges. The set of stable states can be determined similar to (19) by ensuring that the uncontrolled successor state is in $\mathbb{T}_k$ independent of the current parameter.

VI. Numerical Example

This section consists of a numerical example, demonstrating the application of the proposed method and comparing it to robust MPC and nonlinear MPC. We consider the following nonlinear system:

$$x_{k+1}^{[1]} = 0.85 x_k^{[1]} + u_k\,, \tag{26}$$

$$x_{k+1}^{[2]} = (0.25 - 0.55(0.1 x_k^{[2]})^2) x_k^{[1]} + 0.65 x_k^{[2]} + (-1 + 2(0.1 x_k^{[2]})^2) u_k\,, \tag{27}$$

under the constraints

$$-0.5 \leq u_k \leq 1, \qquad \begin{bmatrix} -10 \\ -10 \end{bmatrix} \leq x_k \leq \begin{bmatrix} 8 \\ 8 \end{bmatrix}. \tag{28}$$

This system can be modelled as an LPV system (1) by defining the following scheduling parameter

$$\theta_k = \begin{bmatrix} 1 - (0.1 x_k^{[2]})^2 & (0.1 x_k^{[2]})^2 \end{bmatrix}^T, \tag{29}$$

resulting in the parameter-varying system matrices (4) with the vertices

$$A_1 = \begin{bmatrix} 0.85 & 0 \\ 0.25 & 0.65 \end{bmatrix}, \qquad B_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \tag{30a}$$

$$A_2 = \begin{bmatrix} 0.85 & 0 \\ -0.3 & 0.65 \end{bmatrix}, \qquad B_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \tag{30b}$$

The Multi-Parametric Toolbox (MPT) and YALMIP were used to compute the control laws, [25], [22]. The weight matrices

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad R = 0.01\,, \quad P = Q \tag{31}$$

and a prediction horizon of $N = 3$ were chosen. The ∞-norm was used in the cost function.

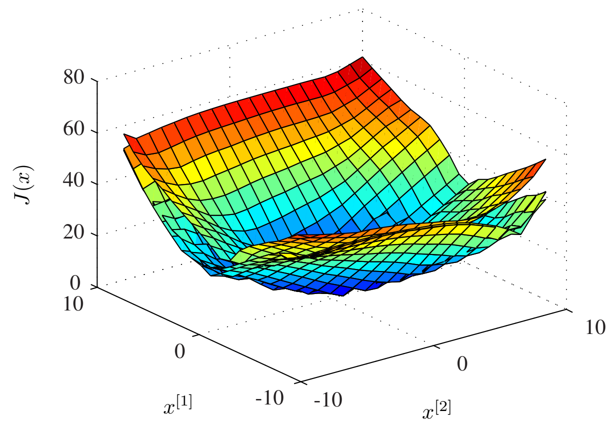

Fig. 2. Actual simulated cost over 40 steps.

Five different controllers for this system were compared in terms of complexity and control performance. By considering the scheduling parameter (29) as an unknown, bounded parametric uncertainty, an explicit robust controller was computed, in the following indicated by `rob`. This robust MPC scheme is presented in detail in [12], where it is derived as the solution to the closed-loop constrained robust optimal (CL-CROC) problem. It follows a similar DP approach, where the control law (5) is parameter-independent, $u_k = \mu(x_k)$.

Following the proposed procedure in this paper, three explicit LPV controller were computed, with (i) an affine parametrization in the scheduling parameter and the Pólya degree 2 – `aff2`, (ii) an affine parametrization in the parameter and a Pólya degree 10 – `aff10`, and (iii) a quadratic parametrization and a Pólya degree of 2 – `qu2`. Finally, the truly optimal solution, based on solving the optimal control problem for the nonlinear model online with the global branch-and-bound based solver in YALMIP, was also used – `nl`. Stability of the closed-loop systems under explicit LPV control was verified for the whole feasible space (28) following the reachability analysis presented in Section V.

All control laws were tested in simulations by controlling the system from 400 initial points, uniformly distributed over the feasible space. Figure 2 shows the actual simulated costs accumulated over 40 steps. It can be observed that the robust control yields a higher cost, while the other control laws are in about the same range. This observation is quantified in Table I, which reveals that there is virtually no difference in performance between the LPV controllers `aff2` and `aff10`. The quadratic controller `qu2` shows a slightly better behavior. One has to mention here that we lack a guarantee of obtaining a better accumulated cost when using a less conservative approximation, since we optimize worst-case performance over a finite horizon.

Table I also shows the complexity of the explicit control laws. There is an increase in complexity from the robust to the affine parametrization and then to the quadratic para-

Controller:	`rob`	`aff2`	`aff10`	`qu2`
No. of regions:	31	53	55	117
Avg. cost increase:	23.3 %	0.4 %	0.4 %	0.2 %

TABLE I

COMPLEXITY OF THE EXPLICIT CONTROL LAWS AND AVERAGE COST INCREASE COMPARED TO NONLINEAR MPC.

metrization. This is due to the number of piecewise affine control laws $\mu_k^j(z_k)$ needed to compose the explicit control laws. In this example the choice of the Pólya degree only had a small influence on the complexity of the resulting controllers.

VII. CONCLUSIONS

In this paper, a method was proposed to compute explicit control laws for LPV systems, linear time-discrete systems with parameter-varying matrices. A parameterization of the input in terms of the scheduling parameter was used in a dynamic programming approach similar to min-max MPC for uncertain systems. This enables the advantages of explicit MPC – control under constraint satisfaction for systems with high sampling rate – for the class of LPV systems.

It was shown in a comparison with robust and nonlinear control that the exploitation of the scheduling parameter increases control performance and nearly reaches the performance of nonlinear MPC.

A drawback of explicit control laws is that the number of controller regions grows exponentially with the prediction horizon and the states. As the suggested approach is based on multi-parametric programming, it suffers from this drawback and thus one future direction of research will be the development of approximate control laws with reduced complexity.

Another direction of investigations will concern the final step cost function. Desirable would be a scheme analogue to the USS, which guarantees optimality of the control law independent of the current scheduling parameter.

REFERENCES

[1] J. Shamma and M. Athans, "Analysis of gain scheduled control for nonlinear plants," *IEEE Trans. on Automatic Control*, vol. AC-35, pp. 898 – 907, 1990.

[2] P. Apkarian, P. Gahinet, and G. Becker, "Self-scheduled H_∞ control of linear parameter-varying systems: A design example," *Automatica*, vol. 31, no. 9, pp. 1251 – 1261, 1995.

[3] J. Shamma and M. Athans, "Guaranteed properties of gain scheduled control for linear parameter-varying plants," *Automatica*, vol. 27, pp. 559 – 564, 1991.

[4] Y. Lu and Y. Arkun, "Quasi-min-max MPC for LPV systems," *Automatica*, vol. 36, pp. 527 – 540, 2000.

[5] B. Pluymers, J. Rossiter, J. Suykens, and B. De Moor, "Interpolation based MPC for LPV systems using polyhedral invariant sets," in *Proc. of the American Control Conference*, Portland, OR, USA, June 2005.

[6] L. Chisci, P. Falugi, and G. Zappa, "Gain-scheduling MPC for nonlinear systems," *Int. Journal of Robust and Nonlinear Control*, vol. 13, pp. 295 – 308, 2003.

[7] B. Bank, J. Guddat, D. Klatte, B. Kummer, and K. Tammer, *Non-Linear Parametric Optimization*. Berlin: Akademie-Verlag, 1982.

[8] A. Bemporad and M. Morari, "Control of systems integrating logics, dynamics, and constraints," *Automatica*, vol. 35, no. 5, pp. 407 – 427, 1999.

[9] E. Pistikopoulos, V. Dua, N. Bozinis, A. Bemporad, and M. Morari, "On-line Optimization via Off-line Parametric Optimization Tools," in *International Symposium on Process Systems Engineering*, Keystone, USA, July 2000, pp. 183–188.

[10] A. Bemporad, F. Borrelli, and M. Morari, "Model predictive control based on linear programming – the explicit solution," *IEEE Transactions on Automatic Control*, vol. 47, no. 12, pp. 1974 – 1985, December 2002.

[11] F. Borrelli, *Constrained Optimal Control of Linear and Hybrid Systems*, ser. Lecture Notes in Control and Information Sciences. Springer-Verlag, 2003, vol. 290.

[12] A. Bemporad, F. Borrelli, and M. Morari, "Min-max control of constrained uncertain discrete-time linear systems," *IEEE Transactions on Automatic Control*, vol. 48, no. 9, pp. 1600 – 1606, September 2003.

[13] V. Sakizlis, N. Kakalis, V. Dua, J. Perkins, and E. Pistikopoulos, "Design of robust model-based controllers via parametric programming," *Automatica*, vol. 40, pp. 189 – 201, 2004.

[14] T. Besselmann, J. Löfberg, and M. Morari, "Explicit model predictive control for systems with parameter-varying state transition matrix," in *IFAC World Congress*, Seoul, Korea, 2008.

[15] M. Barić, S. Raković, T. Besselmann, and M. Morari, "Max-min optimal control of constrained discrete-time systems," *IFAC World Congress*, 2008.

[16] L. Silverman and H. Meadows, "Controllability and observability in time-variable linear systems," *SIAM Journal on Control*, vol. 5, no. 1, pp. 64 – 73, 1967.

[17] G. Balas, J. Bokor, and Z. Szabó, "Invariant subspaces for LPV systems and their applications," *IEEE Trans. on Automatic Control*, vol. 48, no. 11, pp. 2065 – 2069, Nov. 2003.

[18] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific, 1995, vol. one.

[19] J. Rossiter, B. Pluymers, J. Suykens, and B. De Moor, "A multi parametric quadratic programming solution to robust predictive control," in *IFAC World Congress*, Prague, Czech Republic, March 2005.

[20] G. Pólya, "Über positive Darstellung von Polynomen," *Vierteljahrschrift d. naturforschenden Gesellschaft in Zürich*, vol. 73, pp. 141 – 145, 1928, reprinted in: Collected Papers, Volume 2, 309 – 313, Cambridge: MIT Press, 1974.

[21] V. Powers and B. Reznick, "A new bound for Pólya's theorem with applications to polynomial positive on polyhedra," *Journal of Pure and Applied Algebra*, vol. 164, pp. 221 – 229, 2001.

[22] J. Löfberg, "Modeling and solving uncertain optimization problems in YALMIP," *IFAC World Congress*, 2008.

[23] F. Blanchini, "Ultimate boundedness control for uncertain discrete-time systems via set-induced lyapunov functions," *IEEE Trans. on Automatic Control*, vol. 39, no. 2, pp. 428 – 433, February 1994.

[24] ——, "Nonquadratic lyapunov functions for robust control," *Automatica*, vol. 31, no. 3, pp. 451 – 461, March 1995.

[25] M. Kvasnica, P. Grieder, M. Baotic, and M. Morari, "Multi-Parametric Toolbox (MPT)," in *Hybrid Systems: Computation and Control*, Mar. 2004, pp. 448–462.