Automatica 47 (2011) 2292–2297

Contents lists available at SciVerse ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

Stabilizing polynomial approximation of explicit MPC[☆]

Michal Kvasnica[a,1], Johan Löfberg[b], Miroslav Fikar[a]

[a] Faculty of Chemical and Food Technology, Slovak University of Technology in Bratislava, 812 37 Bratislava, Slovakia

[b] Division of Automatic Control, Department of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden

ARTICLE INFO

Article history:
Received 23 August 2010
Received in revised form
9 February 2011
Accepted 14 April 2011
Available online 1 September 2011

Keywords:
Asymptotic stability
Predictive control
Function approximation

ABSTRACT

A given explicit piecewise affine representation of an MPC feedback law is approximated by a single polynomial, computed using linear programming. This polynomial state feedback control law guarantees closed-loop stability and constraint satisfaction. The polynomial feedback can be implemented in real time even on very simple devices with severe limitations on memory storage.

1. Introduction

In explicit Model Predictive Control (Bemporad, Morari, Dua, & Pistikopoulos, 2002), parametric programming (Borrelli, 2003) is used to construct a function μ which maps state measurements x onto the optimal control inputs. Provided such a μ exists, real-time implementation of MPC in Receding Horizon fashion (RHMPC) boils down to a mere function evaluation. For a rich class of MPC setups, μ can be shown to be a piecewise affine (PWA) function defined over $N_{\mathcal{R}}$ polytopic regions. The main practical limitation, however, is that the number of regions grows quickly with problem size, having negative impact on the required memory storage and processing power. The number of regions of μ can be reduced e.g. by move blocking (Tøndel & Johansen, 2002), by model reduction techniques (Hovland, Willcox, & Gravdahl, 2008), or by relaxing optimality (Bemporad & Filippi, 2003). Another direction is to a-posteriori simplify the regions either by merging (Geyer, Torrisi, & Morari, 2008; Kvasnica & Fikar, 2010), by replacing them by hyperboxes (Johansen & Grancharova, 2003) or by simplices (Hovd, Scibilia, Maciejowski, & Olaru, 2009). Evaluation of μ for a particular value of x can be simplified by organizing the regions into a binary search tree (Tøndel, Johansen, & Bemporad, 2003), or by building a lattice representation (Wen, Ma, & Ydstie, 2009) of the PWA function μ. A common denominator of all referenced approaches is that they lead to a simpler (sub)optimal RHMPC feedback $\tilde{\mu}$, which still is a PWA function. As a consequence, although a remarkable reduction of complexity can be achieved in certain cases, the memory footprint of the approximation $\tilde{\mu}$ still typically exceeds ten kilobytes. In this work, we aim at simplifying the RHMPC in such a way that it can easily be implemented on typical industrial hardware platforms, such as programmable logic controllers, which usually only provide 2–8 kB of memory.

We propose to remove all regions of μ completely and to approximate it by a single polynomial feedback $\tilde{\mu}(x)$ of an *a-priori fixed* degree such that closed-loop stability and constraint satisfaction are preserved. The approach is applicable not only to linear systems, but also covers switched affine systems which belong to the class of hybrid systems (Bemporad & Morari, 1999). Building upon our previous work (Kvasnica, Christophersen, Herceg, & Fikar, 2008; Kvasnica, Löfberg, Herceg, Čirka, & Fikar, 2010), the approximation is performed in two steps. First, a parameterization of a set of stabilizing controllers, referred to as the *stability tube* (Christophersen, 2007), is obtained using basic computational geometry tools. Subsequently, we show how to search for the coefficients of $\tilde{\mu}$ by solving a single linear program (LP). If the LP is feasible, the polynomial control law is guaranteed to reside in the stability tube, and hence it is closed-loop stabilizing and satisfies constraints for all time.

The key advantage is that the memory footprint of the approximate feedback $\tilde{\mu}$ is minute compared to the storage of μ. In particular, for the type of problems considered here, the total

☆ The authors are pleased to acknowledge the financial support of the Scientific Grant Agency of the Slovak Republic under the grants 1/0079/09 and 1/0095/11 and of the Slovak Research and Development Agency under the contracts No. VV-0029-07 and No. LPP-0092-07. This paper was not presented at any IFAC meeting. This paper was recommended for publication in revised form by Associate Editor Martin Guay under the direction of Editor Frank Allgöwer.

E-mail addresses: michal.kvasnica@stuba.sk (M. Kvasnica), johanl@isy.liu.se (J. Löfberg), miroslav.fikar@stuba.sk (M. Fikar).

[1] Tel.: +421 2 59325352; fax: +421 2 59325340.

0005-1098/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2011.08.023

storage for $\tilde{\mu}$ is roughly equal to the footprint of a single region of μ. It follows that the overall memory requirements are reduced $N_{\mathcal{R}}$ times. The price to be paid is the inherent loss of optimality. Moreover, certain assumptions on the shape of the stability tube have to be imposed in order to formulate the search for coefficients of $\tilde{\mu}$ as a single LP. Because of this, and since the LP-based search is based on sufficient conditions, it does not have to be always successful.

Compared to our previous work (Kvasnica et al., 2008, 2010), we report a detailed complexity analysis of the overall design procedure. A large case study is provided to illustrate how the computation scales with increasing problem size and to assess the overall success rate. More importantly, new ideas for reducing the size of the LP problem are presented which extend the applicability of our approach to larger problems.

2. Preliminaries

The set of non-negative real numbers is denoted by $\mathbb{R}_{\geq 0}$. Interior of a set Ω is $\operatorname{int}(\Omega)$. We call a collection of polytopes $\{\mathcal{R}_i\}_{i=1}^{N_{\mathcal{R}}}$ the *partition* of a set Ω if $\Omega = \bigcup_{i=1}^{N_{\mathcal{R}}} \mathcal{R}_i$, and $\operatorname{int}(\mathcal{R}_i) \cap \operatorname{int}(\mathcal{R}_j) = \emptyset$ for all $i \neq j$. Each polytope $\mathcal{R}_i$ will be referred to as a *region* of the partition. A function $\mu : \mathbb{R}^{n_x} \to \mathbb{R}^{n_u}$ with domain $\Omega \subseteq \mathbb{R}^{n_x}$ is called Piecewise Affine (PWA) over polytopes if $\{\mathcal{R}_i\}_{i=1}^{N_{\mathcal{R}}}$ is a partition of Ω and $\mu(x) := K_i x + L_i \; \forall x \in \mathcal{R}_i,\ i = 1, \ldots, N_{\mathcal{R}}$.

We consider well-posed (Bemporad & Morari, 1999), stabilizable PWA systems in discrete time $x_{t+1} = f_{\mathrm{PWA}}(x_t, u_t)$, composed of finitely many local affine dynamics, each valid in a polytope $\mathcal{D}_d \subseteq \mathbb{R}^{n_x}$:

$$f_{\mathrm{PWA}}(x_t, u_t) := A_d x_t + B_d u_t + a_d \quad \text{if } x_t \in \mathcal{D}_d, \tag{1}$$

where $x_t \in \mathbb{R}^{n_x}$ are the states and $u_t \in \mathbb{R}^{n_u}$ the inputs. The task is to control the PWA system (1) toward the origin (which is assumed to be an equilibrium of (1)) while fulfilling state and input constraints for all time, i.e. $x_t \in \mathcal{X},\ u_t \in \mathcal{U},\ \forall t \geq 0$, where $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ and $\mathcal{U} \subseteq \mathbb{R}^{n_u}$ are assumed to be non-empty polytopic sets containing the origin in their respective interiors.

We define for the PWA system (1) the *constrained finite time optimal control* (CFTOC) problem

$$J_N^*(x_t) = \min_{U_N} \ \ell_N(x_{t+N}) + \sum_{k=0}^{N-1} \ell(x_{t+k}, u_{t+k}) \tag{2a}$$

$$\text{s.t.} \begin{cases} x_{t+k+1} = f_{\mathrm{PWA}}(x_{t+k}, u_{t+k}), \\ u_{t+k} \in \mathcal{U}, \qquad x_{t+k} \in \mathcal{X}, \qquad x_{t+N} \in \mathcal{X}_f, \end{cases} \tag{2b}$$

where x_{t+k} is the future evolution of (1) over a prediction horizon N, given the initial condition x_t and the vector of future control inputs $U_N := [u_t^T, \ldots, u_{t+N-1}^T]^T$. $\mathcal{X}_f \subseteq \mathcal{X}$ is a polytopic terminal set with $\mathbb{0}_{n_x} \in \mathcal{X}_f$, $\ell_N(x_{t+N}) = \|Q_N x_{t+N}\|_p$ is the terminal penalty, and $\ell(x_{t+k}, u_{t+k}) = \|Q_x x_{t+k}\|_p + \|Q_u u_{t+k}\|_p$ is the stage cost. It is assumed that $p \in \{1, \infty\}$ in (2a). For problems of modest size it is possible to characterize the RHMPC feedback law $\mu : \Omega \to \mathcal{U}$ and the optimal value function $J_N^* : \Omega \to \mathbb{R}_{\geq 0}$ explicitly as PWA functions of x_t (Bemporad et al., 2002; Borrelli, 2003). Here, $\Omega := \{x_t \mid \exists u_t, \ldots, u_{t+N-1} \text{s.t. (2b) hold}\}$, and it is partitioned into $N_{\mathcal{R}}$ polytopic regions $\mathcal{R}_i$.

Assumption 2.1. The RHMPC feedback $\mu(x_t)$ is closed-loop stabilizing, feasible for all time (Christophersen, 2007) and a PWA Lyapunov function $V : \Omega \to \mathbb{R}_{\geq 0}$ for the closed-loop system $f^{\mathrm{CL}} := f_{\mathrm{PWA}}(x_t, \mu(x_t))$ exists $\forall x_t \in \Omega$ and is given.

This is not a restricting requirement but rather the aim of most (if not all) control strategies. We remark that if $N, Q_x, Q_u, Q_N, \mathcal{X}_f$ are chosen as in Baotić, Christophersen, and Morari (2006), then $\mu(\cdot)$ satisfies Assumption 2.1 and $V := J_N^*$ is a Lyapunov function.

Theorem 2.2 (*Lazar, Munoz de la Pena, Heemels, & Alamo, 2008*). *Let Ω be a bounded positively invariant set with $\mathbb{0}_{n_x} \in \operatorname{int}(\Omega)$ and let $\underline{\beta}(\cdot)$ and $\overline{\beta}(\cdot)$ be $\mathcal{K}_\infty$-class functions. Then if there exists function $V : \Omega \to \mathbb{R}_{\geq 0}$ with $V(\mathbb{0}_{n_x}) = 0$, bounded by $\underline{\beta}(\|x\|) \leq V(x) \leq \overline{\beta}(\|x\|)$, and satisfying $V(f^{\mathrm{CL}}(x)) \leq \gamma V(x)$ for some $\gamma \in [0, 1)$ and for all $x \in \Omega$, then the closed-loop system f^{CL} is asymptotically stable in Ω.*

The freedom in γ allows one to find a set of stabilizing controllers which render the function V a control Lyapunov function. Such sets are denoted as *stability tubes* (Christophersen, 2007):

$$\mathcal{S}(V, \gamma) := \left\{ \begin{bmatrix} x \\ u \end{bmatrix} \mid u \in \mathcal{U},\ x \in \Omega,\ f(x, u) \in \Omega, \right. \\ \left. V(f(x, u)) \leq \gamma V(x) \right\}. \tag{3}$$

For the type of PWA systems (1), PWA Lyapunov functions V, and fixed γ, the tube can be computed explicitly using reachability analysis (Christophersen, 2007, Ch. 10.4) and represented as a (possibly non-convex) union of polytopes. To see this, note that for each feasible transition from region $\mathcal{R}_i$ to region $\mathcal{R}_j$ for which the value of V decreases, (3) is a polytope $\mathcal{S}_{i,j}$ in the $x - u$ space. The whole stability tube is then given by $\mathcal{S}(V, \gamma) := \bigcup_{i=1}^{N_{\mathcal{R}}} \mathcal{S}_i$, where $\mathcal{S}_i := \bigcup_{j=1}^{N_{\mathcal{R}}} \mathcal{S}_{i,j},\ i = 1, \ldots, N_{\mathcal{R}}$.

3. Main results

We aim at approximating a given RHMPC control law μ by a single multivariate polynomial $\tilde{\mu}$ of *pre-specified* degree δ:

$$\tilde{\mu}(x) = \alpha_1 x + \alpha_2 x^2 + \cdots + \alpha_\delta x^\delta. \tag{4}$$

Here, $\alpha_i \in \mathbb{R}^{n_u \times n_x},\ i = 1, \ldots, \delta$, are the coefficients to be determined, and x^i is the element-wise i-th power of a vector $x \in \mathbb{R}^{n_x}$, i.e. $x^i = [x_1^i, x_2^i, \ldots, x_{n_x}^i]^T$. Note that in a multi-input case with $n_u > 1$, (4) is a vector-valued polynomial. The constant offset α_0 is not considered in (4) since $\tilde{\mu}(\mathbb{0}_{n_x}) = \mathbb{0}_{n_u}$ must hold to attain stability. Formally, we aim at solving the following problem.

Problem 3.1. Find the coefficients $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_\delta\}$ of the polynomial state-feedback law (4) of fixed degree δ such that $\tilde{\mu} \approx \mu$ asymptotically stabilizes the PWA system (1) to the origin while fulfilling state and input constraints for all time.

To solve this problem, we exploit the inherent freedom of the Lyapunov function V, captured by its stability tube.

Theorem 3.2 (*Christophersen, 2007*). *Let the stability tube $\mathcal{S}(V, \gamma)$ be given. Then every control law $\tilde{\mu}(x)$ fulfilling $\left[\begin{smallmatrix} x \\ \tilde{\mu}(x) \end{smallmatrix}\right] \in \mathcal{S}(V, \gamma)$ asymptotically stabilizes the system $x^+ = f_{\mathrm{PWA}}(x, \tilde{\mu}(x))$ for all $x \in \Omega$ to the origin.*

Remark 3.3. The concept of stability tubes does not require that the function V originates as a solution of the MPC problem (2). In fact, the tube can be constructed for an arbitrary feedback law which admits a PWA[2] Lyapunov function on Ω. It follows that the presented procedure can be applied to approximate arbitrary feedback laws with this property.

In the sequel we show that, given a stability tube $\mathcal{S}(V, \gamma)$, the polynomial $\tilde{\mu}$ satisfying $\left[\begin{smallmatrix} x \\ \tilde{\mu}(x) \end{smallmatrix}\right] \in \mathcal{S}(V, \gamma),\ \forall x \in \Omega$ can be found by solving a single linear program under the following assumption.

[2] For piecewise quadratic Lyapunov functions the tube can no longer be represented as a union of polytopes, in general.

Assumption 3.4. For a given Lyapunov function V there exists a $\gamma \in [0, 1)$ for which:

A1: a full-dimensional stability tube $\mathcal{S}(V,\gamma) := \bigcup_{i=1}^{N_{\mathcal{R}}} \mathcal{S}_i$ exists;
A2: for each $i = 1, \ldots, N_{\mathcal{R}}$ either $\mathcal{S}_i := \cup_j \mathcal{S}_{i,j}$ is a convex polytope, or an inner polytopic approximation $\mathcal{S}_i \subseteq \cup_j \mathcal{S}_{i,j}$ exists such that $\mathrm{proj}_x \mathcal{S}_i = \mathcal{R}_i$;
A3: the union $\cup_i \mathcal{S}_i$ is connected.

The existence of $\mathcal{S}(V,\gamma)$ hints at the existence of control laws, other than μ, which would provide closed-loop stability and constraint satisfaction for all time. Connectivity is implied by the objective of approximating μ by a *single continuous* polynomial valid over the whole domain $\mathrm{dom}(\tilde{\mu}) = \Omega$. Finally, convexity (and hence uniqueness) is dictated by the desire of being able to perform the approximation in a computationally efficient manner. If A2 does not hold, $\tilde{\mu}$ can still be found by solving a combinatorial problem. If A3 is violated, the remedy would be to approximate independently each connected part of the tube, giving rise to a piecewise polynomial type of approximation.

Under this assumption, the tube consists of $N_{\mathcal{R}}$ polytopes in the state-input space:

$$\mathcal{S}_i := \left\{ \begin{bmatrix} x \\ u \end{bmatrix} \;\middle|\; \begin{bmatrix} S_i^x & S_i^u \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \leq S_i^0 \right\}. \tag{5}$$

We remark that the whole tube $\mathcal{S}(V,\gamma) := \cup_i \mathcal{S}_i$ is not required to be convex. Define, for each $i = 1, \ldots, N_{\mathcal{R}}$, a set of polynomials

$$p_i(\boldsymbol{\alpha}, x) := S_i^0 - S_i^x x - S_i^u \tilde{\mu}(x), \tag{6}$$

where the cardinality of $p_i(\cdot)$ is equal to the number of constraints of the i-th element of the stability tube, i.e. the number of rows of S_i^0. Then we get the following straightforward result.

Lemma 3.5. *Let a stability tube $\mathcal{S}(V,\gamma)$ satisfying Assumption 3.4 be given. If there exist coefficients $\alpha_1, \ldots, \alpha_\delta$ of $\tilde{\mu}$ as in (4) such that*

$$p_i(\boldsymbol{\alpha}, x) \geq 0, \quad \forall x \in \mathcal{R}_i,\ i = 1, \ldots, N_{\mathcal{R}}, \tag{7}$$

then $\tilde{\mu}$ solves Problem 3.1.

Proof. First note that (7) with $p_i(\boldsymbol{\alpha}, x)$ as in (6) is equivalent, for a fixed i, to (5) with $u = \tilde{\mu}(x)$. Therefore, if (7) admits a solution, then $\begin{bmatrix} x \\ \tilde{\mu}(x) \end{bmatrix} \in \mathcal{S}_i \forall x \in \mathcal{R}_i$. Hence if (7) holds for all $i = 1, \ldots, N_{\mathcal{R}}$, it follows from Theorem 3.2 that $\tilde{\mu}$ provides closed-loop stability and constraint satisfaction for all $x \in \Omega$. □

Lemma 3.5 suggests that finding $\tilde{\mu}$ of the form (4) as a solution to Problem 3.1 can be cast as finding the coefficients $\alpha_1, \ldots, \alpha_\delta$ such that polynomials $p_i(\boldsymbol{\alpha}, x)$ are non-negative over corresponding regions. There is a subtle, yet very important issue which makes solving problem (7) far from straightforward: even for a fixed i, all polynomials $p_i(\cdot)$ associated to region $\mathcal{R}_i$ must be non-negative *for all* points $x \in \mathcal{R}_i$, not just for *some* of them (e.g. for the vertices of $\mathcal{R}_i$). One approach is to employ the Positivstellensatz and show positivity of polynomials by solving a sum-of-squares problem, as suggested in Kvasnica et al. (2008). However, as documented in Kvasnica et al. (2010), such a procedure is, from a practical point of view, limited to small-scale problems only. A different direction is therefore persuaded here, which is based on the following theorem, originally formulated by Hardy, Littlewood, and Pólya (1952) to show strict positivity of polynomials and later extended to the non-strict case by Mok and To (2008).

Theorem 3.6 (*Pólya's Theorem*)**.** *If a homogeneous polynomial $p_i(\boldsymbol{\alpha}, x)$ is non-negative over a unit simplex, then all the coefficients of the extended polynomial $p_i^M(\boldsymbol{\alpha}, x) = p_i(\boldsymbol{\alpha}, x) \cdot \left(\sum_{j=1}^{n_x} x_j\right)^M$ are non-negative for a sufficiently large Pólya degree M.*

Remark 3.7. Search for the coefficients $\boldsymbol{\alpha}$, such that $p_i^M(\boldsymbol{\alpha}, x)$ is non-negative over a simplex can be performed by using the more obvious reverse of Pólya's theorem, i.e. non-negative coefficients of the extended polynomial imply its non-negativity over the whole simplex.

Corollary 3.8. *Given a symbolic representation of coefficients of $p_i^M(\boldsymbol{\alpha}, x)$, the coefficients $\boldsymbol{\alpha}$ of $\tilde{\mu}$ can be found by solving a linear program. To see this, observe that $\boldsymbol{\alpha}$ enters (6) in a linear fashion per definition of $\tilde{\mu}$ as in (4). All constraints in (7) are therefore linear in $\boldsymbol{\alpha}$.*

Note, however, that Theorem 3.6 is not directly applicable to find $\boldsymbol{\alpha}$ from (7) as $\mathcal{R}_i$ are not unit simplices with $\mathbb{0}_{n_x} \in \mathcal{R}_i$, in general. Therefore, we propose to represent the polytopic regions in their equivalent vertex representation, i.e. by

$$\mathcal{R}_i = \left\{ x \mid x = \sum_{j=1}^{|\mathcal{V}_i|} \lambda_j [\mathcal{V}_i]_j,\ \lambda \in \Lambda_i \right\}, \tag{8a}$$

$$\Lambda_i = \left\{ \lambda \mid 0 \leq \lambda_j \leq 1,\ \sum_{j=1}^{|\mathcal{V}_i|} \lambda_j = 1 \right\}. \tag{8b}$$

Here, $\mathcal{V}_i$ are the vertices the i-th region, $|\mathcal{V}_i|$ denotes their cardinality, $[\mathcal{V}_i]_j$ is the j-th vertex of $\mathcal{R}_i$, and $\lambda = [\lambda_1, \ldots, \lambda_{|\mathcal{V}_i|}]$. By substituting for $x = \sum_j \lambda_j [\mathcal{V}_i]_j$ into (6) and (7), we get

$$p_i(\boldsymbol{\alpha}, \lambda) \geq 0, \quad \forall \lambda \in \Lambda_i,\ i = 1, \ldots, N_{\mathcal{R}}. \tag{9}$$

Note that Λ_i in (9) are now $|\mathcal{V}_i|$-dimensional unit simplices and Theorem 3.6 can therefore be applied to find $\boldsymbol{\alpha}$ such that $p_i(\boldsymbol{\alpha}, \lambda)$ is non-negative $\forall \lambda \in \Lambda_i,\ i = 1, \ldots, N_{\mathcal{R}}$. Also note that such change of variables is needed even if all $\mathcal{R}_i$ originally were simplices, since the Pólya's Theorem only applies if $\mathbb{0}_{n_x} \in \mathcal{R}_i$.

We can now state the main result of the paper, which is Theorem 3.9 and Algorithm 1 for calculating values of the coefficients $\alpha_1, \ldots, \alpha_\delta$ of the polynomial feedback law $\tilde{\mu}$ which is an admissible solution to Problem 3.1.

Algorithm 1 Polynomial approximation

INPUT: Optimal RHMPC feedback law μ, PWA Lyapunov function V, scalar $\gamma \in [0, 1)$, degree of the approximation polynomial δ, Pólya degree M.
OUTPUT: Coefficients $\alpha_1, \ldots, \alpha_\delta$ of the polynomial feedback law (4).

1: Obtain the stability tube $\mathcal{S}(V,\gamma)$ per (5).
2: Calculate extremal vertices $\mathcal{V}_i$ of all regions $\mathcal{R}_i$.
3: Formulate polynomials $p_i(\boldsymbol{\alpha}, \lambda)$ per (8)–(9).
4: Homogenize $p_i(\boldsymbol{\alpha}, \lambda)$ by multiplying single monomials by $\left(\sum_{j=1}^{|\mathcal{V}_i|} \lambda_j\right)$ until all monomials have the same degree.
5: Obtain symbolic representation of coefficients c_i^M of Pólya's polynomials $p_i^M(\boldsymbol{\alpha}, \lambda) = p_i(\boldsymbol{\alpha}, \lambda) \cdot \left(\sum_{j=1}^{|\mathcal{V}_i|} \lambda_j\right)^M$.
6: Search for $\boldsymbol{\alpha}$ by solving a linear program:

$$\text{find } \alpha_1, \ldots, \alpha_\delta, \tag{10a}$$

$$\text{s.t. } c_i^M \geq 0, \quad i = 1, \ldots, N_{\mathcal{R}}. \tag{10b}$$

Theorem 3.9. *Let the input arguments of Algorithm 1 satisfy Assumption 2.1 and assume that the tube $\mathcal{S}(V,\gamma)$ computed in Step 1 satisfies Assumption 3.4. If the LP (10a) is feasible, the polynomial feedback law $\tilde{\mu}$ of the form (4) calculated by Algorithm 1 is a solution to Problem 3.1.*

Proof. If (10) is feasible, then, according to Theorem 3.6, polynomials $p_i(\boldsymbol{\alpha}, \lambda)$ are non-negative over corresponding regions $\mathcal{R}_i$. This in turn implies that (7) is satisfied, which, according to Lemma 3.5, shows that $\tilde{\mu}(x)$ belongs to the stability tube $\mathcal{S}(V, \gamma), \forall x \in \Omega$. Therefore, by Theorem 3.2, $\tilde{\mu}$ is guaranteed to be closed-loop stabilizing and feasible for all time. □

Remark 3.10. Algorithm 1 is a non-iterative procedure and therefore it always terminates in a single pass, provided that all of its steps are successful. However, since Theorems 3.2 and 3.6 are only sufficient conditions for the existence of a stabilizing feedback $\tilde{\mu}$, the algorithm may fail to find it even if one exists.

Instead of a pure feasibility objective in (10a), an alternative is to minimize the point-wise distance $|\mu(x_j) - \tilde{\mu}(x)(x_j)|_q$ with $q \in \{1, \infty\}$ over some points x_j (e.g. over the vertices of each $\mathcal{R}_i$). Doing so will let $\tilde{\mu}$ to follow the shape of $\mu(x)$ more tightly, hence mitigating the induced loss of optimality. Another approach is to aim for low-order polynomials. This can be done in three ways: (i) minimize the ℓ_1 norm of $\boldsymbol{\alpha}$, which tends to give sparse solutions; (ii) use bisection in conjunction with Algorithm 1; or (iii) minimize the number of non-zero coefficients to a global minimum by solving a mixed-integer version of (10).

Example 3.11. Consider the following open-loop unstable PWA system (Kvasnica et al., 2008):

$$x_{t+1} = \begin{cases} 6/5x_t - 2u_t & \text{if } x_t \geq 0, \\ -4/5x_t + u_t & \text{otherwise}, \end{cases} \tag{11}$$

with $u_t \in [-1, 1]$ and $x_t \in [-4, 4]$. With the choice of $p = 1$, $Q_x = 1$, $Q_u = 1$, $N = \infty$ in (2) we obtain a stabilizing feedback μ and a Lyapunov function J_N^* defined over 6 regions. The stability tube $\mathcal{S}(V, \gamma)$ for $V := J_N^*$ and $\gamma = 0.7$, the optimal RHMPC feedback $\mu(x_t)$, and its polynomial approximations $\tilde{\mu}$ of degrees $\delta = 3, 5, 7$ computed by Algorithm 1 are shown in Fig. 1.

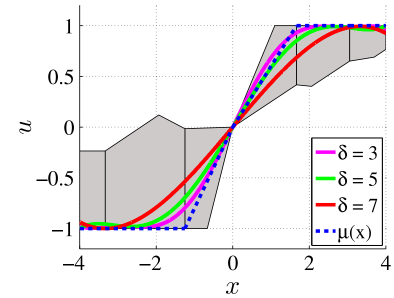


Fig. 1. Stability tube $\mathcal{S}(V, \gamma)$ for $\gamma = 0.7$ (gray sets), optimal control law μ (blue dashed line), and stabilizing polynomials $\tilde{\mu}$ of different degrees δ. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4. Complexity analysis

4.1. Complexity of Algorithm 1

Computation of stability tubes $\mathcal{S}(V, \gamma)$ in Step 1 can be done in $\mathcal{O}(N_{\mathcal{R}}^2)$ time, since all possible transitions between various regions have to be investigated. The LP in Step 6 has $n_u n_x \delta$ variables (the coefficients $\alpha_1, \ldots, \alpha_\delta$) and $\mathcal{O}(N_c)$ constraints. Here, $N_c = \sum_{i=1}^{N_{\mathcal{R}}} N_{c,i}$ is the total number of coefficients of Pólya polynomials $p_i^M(\boldsymbol{\alpha}, \lambda)$, with $N_{c,i} = \binom{\delta_p + |\mathcal{V}_i| - 1}{\delta_p}$ and $\delta_p = \delta + M$, where $|\mathcal{V}_i|$ is the number of vertices of the i-th region.

Remark 4.1. The number of Pólya's coefficients, and hence the number of constraints of the LP (10), grows quickly with the number of states n_x. In the most general case, $|\mathcal{V}_i| = \mathcal{O}(2^{n_x})$. This is the main bottleneck of the presented procedure. One way to mitigate such a quick growth is to triangulate the regions $\mathcal{R}_i$. Although this will increase the total number of regions to, at most, $\mathcal{O}\left(\sum_{i=1}^{N_{\mathcal{R}}} |\mathcal{V}_i|^{\lceil n_x/2 \rceil}\right)$, the gained advantage is that $|\mathcal{V}_i|$ stays fixed at $n_x + 1, \forall i$. From numerical experiments, we have observed that employing triangulation reduces the total number of constraints in (10) by a factor of 5, on average.

Remark 4.2. Another option to reduce the size of the linear program (10) is to eliminate the redundant constraints. Full redundancy elimination would require solving $\mathcal{O}(N_c)$ copies of the LP (10), which clearly is not an option. Note, however, that a valid solution to (10) has to be non-negative due to Theorem 3.6 and Remark 3.7. Therefore, eliminating from (10b) those constraints where all constant multipliers in c_i^M are non-negative will not affect feasibility. Numerical examples suggest that such a simple elimination reduces the number of constraints of (10) by a factor of 2, on average.

4.2. On-line complexity

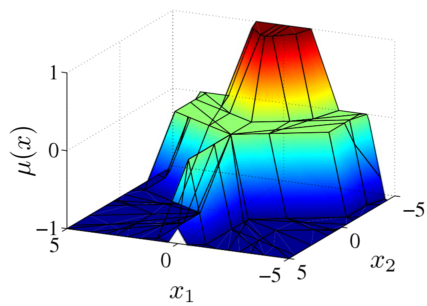
Implementing $\tilde{\mu}$ in a feedback arrangement reduces to a mere evaluation of the polynomial for a given x. Since the polynomial continuously covers the whole state-space of interest, no region search is necessary. Using *Horner's* scheme (Eve, 1964), $\tilde{\mu}$ can be evaluated[3] by at most $1/2 n_u n_x (3\delta + 5)$ FLOPS. Storing the coefficients $\alpha_1, \ldots, \alpha_\delta$ consumes $\delta n_u n_x$ floating point numbers. On the other hand, evaluating the optimal feedback law μ via a binary search tree (Tøndel et al., 2003) requires $\mathcal{O}(\log_2 N_{\mathcal{R}})$ FLOPS and the tree consumes $\mathcal{O}(N_{\mathcal{R}}(n_x + n_u))$ memory elements. Complexity of the lattice representation (Wen et al., 2009), both in terms of runtime and memory, is $\mathcal{O}(N_{\mathcal{U}}^2)$ where $N_{\mathcal{U}}$ denotes the number of unique feedback laws.

5. Examples

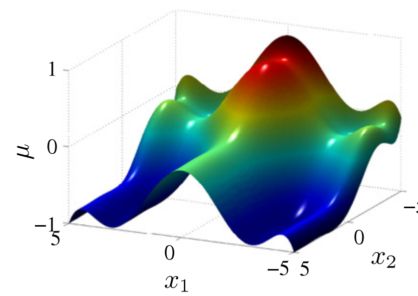
5.1. Standard PWA benchmark

Consider the following PWA system with 2 states and one input, introduced in Bemporad and Morari (1999): $x^+ = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$, where the value of θ switches depending on the value of the first element of the state vector: $\theta = -\pi/3$ if $x_1 \leq 0$, and $\theta = \pi/3$, otherwise. State constraints $|x_i| \leq 5$, $i = 1, 2$ are assumed, along with input bounds $|u| \leq 1$. Even though the system is open-loop stable, a controller is needed to guarantee constraint satisfaction for all time. The explicit RHMPC feedback law μ was constructed by solving (2) with $Q_x = \mathbb{1}$, $Q_u = 1$, $p = \infty$ and $N = \infty$, and consists of 112 regions shown in Fig. 2(a). We have then applied Algorithm 1 to find approximate feedbacks $\tilde{\mu}$ of degrees $\delta = 1, \ldots, 7$. The stability tube in Step 1 was computed for $\gamma = 0.99$ and it satisfied Assumption 3.4. The polynomial of degree 7 is shown in Fig. 2(b). To assess the induced loss of optimality, we have analyzed closed-loop profiles of states and inputs. The performance degradation is given by $\Delta_J := (J^* - \tilde{J})/J^*$, where J^* is the value of (2a) for a closed-loop profile obtained by applying the optimal feedback $\mu(x)$, while $\tilde{J}$ is the cost of the closed-loop evolution driven by $u = \tilde{\mu}(x)$. The average values of Δ_J over 1000 equidistantly spaced initial conditions are reported in Table 1.

[3] If only fixed-point arithmetics is available, evaluation can be done as in Brisebarre, Chevillard, Ercegovac, Muller, and Torres (2008).



(a) RHMPC feedback μ.



(b) Polynomial $\tilde{\mu}$ of degree 7.

Fig. 2. Optimal RHMPC feedback law and its approximation.

Table 1
Performance degradation for various degrees of the polynomial approximation $\tilde{\mu}$.

δ	1	2	3	4	5	6	7
Δ_J (%)	48	46	43	37	36	28	28

5.2. Linear system

Consider the following linear system with 4 states and 1 input:

$$x^+ = \begin{bmatrix} 0.7 & -0.1 & 0.0 & 0.0 \\ 0.2 & -0.5 & 0.1 & 0.0 \\ 0.0 & 0.1 & 0.1 & 0.0 \\ 0.5 & 0.0 & 0.5 & 0.5 \end{bmatrix} x + \begin{bmatrix} 0.1 \\ 1.0 \\ 0.0 \\ 0.0 \end{bmatrix} u,$$

which is subject to constraints $|x_i| \leq 5$, $i = 1, \ldots, 4$, and $|u| \leq 5$. The optimal RHMPC feedback law for $Q_x = 10 \cdot \mathbb{1}_4$, $Q_u = 0.1$, and $N = 3$ has 230 regions in 4D state space. The lowest feasible degree of $\tilde{\mu}$ was $\delta = 1$, leading to the linear feedback $\tilde{\mu}(x) = -0.0715x_1$. It follows that while the on-line implementation of the optimal RHMPC controller would require the storage of 9290 floating point numbers (8140 for describing the 230 regions, and 1628 for the associated feedback laws), the polynomial feedback requires storing exactly one floating point number at the price of a 35% worst-case drop of performance.

This case also illustrates practical consequences of Remarks 4.1 and 4.2. Without any of them applied, the LP (10) for $\delta = 3$ would have $5.8 \cdot 10^6$ constraints, which is above the limit of most LP solvers.[4] Performing triangulation per Remark 4.1 led to $1.8 \cdot 10^6$ inequalities. Further elimination of trivially redundant constraints per Remark 4.2 decreased this figure to $0.9 \cdot 10^6$.

5.3. Random systems

To assess versatility of the presented approach, we have analyzed random PWA systems with 2 dynamics under state constraints $|x| \leq 5$ and input bounds $|u| \leq 1$. Three batches of random systems of various dimensions were considered, with 100 systems in each batch. For each system the optimal RHMPC feedback law μ was computed[5] by solving (2) with $Q_x = \mathbb{1}_{n_x}$, $Q_u = \mathbb{1}_{n_u}$, and $N = 5$. Q_N and $\mathcal{X}_f$ were designed as in Baotić et al. (2006) Subsequently, the stability tubes $\mathcal{S}(V, \gamma)$ were constructed for $V := J_N^*$ and $\gamma = 0.99$. In 62% of the 300 investigated problems the respective stability tubes satisfied Assumption 3.4. Important to notice is that the success rate was 97% when investigating a supplemental batch of 100 random linear systems.

Table 2
Data for random systems.

n_x/n_u	$N_{\mathcal{R}}$	$N_{\mathcal{T}}$	Step 1 (s)	Step 6 (s)	$\delta_{\min}$	Δ_J (%)
2/1	117	215	28	1	1	77
	180	329	54	2	5	23
	251	485	103	1	3	51
	288	552	152	2	6	37
2/2	115	206	62	3	2	81
	197	379	135	1	1	65
	277	522	164	1	5	35
	376	738	328	7	4	42
3/1	270	1258	293	3	3	12
	450	1755	551	1	1	72
	606	2963	829	12	6	41
	834	4278	1453	3	4	65

The tubes were then triangulated according to Remark 4.1 and further processed by Algorithm 1. The runtime of triangulation never exceeded 20 s for any of the investigated examples. Enumeration of vertices in Step 2 never took more than 1 s using MPT (Kvasnica, Grieder, & Baotić, 2004). The LP in Step 6 was formulated by YALMIP (Löfberg, 2004) and solved by CPLEX 12.1 (ILOG, 0000). Only degrees up to 7 were investigated due to practical reasons. The success rate of the LP-based procedure was 81%. No obvious correlation between the number of regions of μ and the required degree δ in (10) was observed. Around 25% of all feasible cases admitted the existence of a linear approximation $\tilde{\mu}$, regardless of n_x and n_u. Higher order approximations with minimal feasible degrees $\delta = 2, \ldots, 6$ appeared with a roughly equal distribution.

A representative selection of the results is reported in Table 2 which shows how the computation scales with increasing problem size. Columns of the tables denote, respectively, state and input dimensions, number of regions $N_{\mathcal{R}}$, number of triangulated regions $N_{\mathcal{T}}$, runtime of construction of the stability tube in Step 1, runtime of the LP in Step 6, minimum degree $\delta_{\min}$ for which the LP was feasible, and the average performance degradation induced by using $\tilde{\mu}$ of the minimal degree. Even though the average performance drop Δ_J might sound large, one has to take into account three facts. First, as discussed previously, performance usually improves if δ is enlarged. Second, and more importantly, design of *any* stabilizing feedback controller for PWA systems is a non-trivial task, even putting optimality aside. Finally, magnitudes of the reported performance drops are similar to what can be achieved by other techniques; see e.g. Bemporad, Oliveri, Poggi, and Storace (2010) and Hovd et al. (2009).

6. Conclusions

We have presented a novel way of deriving simple stabilizing feedback laws for the class of constrained linear and PWA systems.

[4] For instance, the 32-bit version of CPLEX only allows around $2 \cdot 10^6$ constraints.

[5] On a 2.5 GHz CPU and 2 GB of RAM using Matlab R2009a and MPT 2.6.3.

Stability and feasibility of the approximate polynomial controllers are guaranteed by employing the concept of stability tubes, which can be viewed as a parameterization of stabilizing feedback laws. It was illustrated that coefficients of the polynomials can be found by solving a single linear program. Triangulation and a cheap redundancy elimination were proposed as a way to significantly mitigate the size of the LP, hence allowing to process even large problems. Although the presented procedure inherently induces sub-optimality, the synthesized polynomial feedback not only guarantees stability and constraint satisfaction, but also puts very low requirements on its implementation in real time.

Certain restrictions have to be imposed on the shape of stability tubes in order to be able to find the approximation by solving a single LP. Investigation of a large number of random cases showed that a suitable tube was found in 60% of PWA systems, while the success rate is close to 100% when considering linear systems. If the tube has "unfavorable" shape, one would need to resort to a piecewise polynomial nature of the approximation. Although no obvious correlation between the number of elements of the tube and the degree of the approximate polynomial was observed, it cannot be ruled out that higher order polynomials might be necessary to approximate more complex tubes.

References

Baotić, M., Christophersen, F. J., & Morari, M. (2006). constrained optimal control of hybrid systems with a linear performance index. *IEEE Transactions on Automatic Control*, *51*(12), 1903–1919.

Bemporad, A., & Filippi, C. (2003). Suboptimal explicit RHC via approximate multiparametric quadratic programming. *Journal of Optimization Theory and Applications*, *117*(1), 9–38.

Bemporad, A., & Morari, M. (1999). Control of systems integrating logic, dynamics, and constraints. *Automatica*, *35*(3), 407–427.

Bemporad, A., Morari, M., Dua, V., & Pistikopoulos, E. N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, *38*(1), 3–20.

Bemporad, A., Oliveri, A., Poggi, T., & Storace, M. (2010). Synthesis of stabilizing model predictive controllers via canonical piecewise affine approximations. In *Conference on decision and control. CDC. Atlanta, USA* (pp. 5296–5301).

Borrelli, F. (2003). *Lecture notes in control and information sciences*: *Vol. 290*. *Constrained optimal control of linear and hybrid systems*. Springer-Verlag.

Brisebarre, N., Chevillard, S., Ercegovac, M.D., Muller, J.-M., & Torres, S. (2008). An efficient method for evaluating polynomial and rational function approximations. In *Proceedings of the ASAP 2008 conference* (pp. 233–238).

Christophersen, F. J. (2007). *Lecture notes in control and information sciences*: *Vol. 359*. *Optimal control of constrained piecewise affine systems*. Springer Verlag.

Eve, J. (1964). The evaluation of polynomials. *Numerische Mathematik*, *6*, 17–21.

Geyer, T., Torrisi, F. D., & Morari, M. (2008). Optimal complexity reduction of polyhedral piecewise affine systems. *Automatica*, *44*(7), 1728–1740.

Hardy, G. H., Littlewood, J. E., & Pólya, G. (1952). *Inequalities* (second ed.) Cambridge University Press.

Hovd, M., Scibilia, F., Maciejowski, J., & Olaru, S. (2009). Verifying stability of approximate explicit MPC. In *IEEE conference on decision and control. Shanghai, China* (pp. 6345–6635).

Hovland, S., Willcox, K. E., & Gravdahl, J. T. (2008). Explicit MPC for large-scale systems via model reduction. *AIAA Journal of Guidance, Control and Dynamics*, *31*(4).

ILOG, Inc. CPLEX user manual. Gentilly Cedex, France. http://www.ilog.fr/products/cplex/.

Johansen, T. A., & Grancharova, A. (2003). Approximate explicit constrained linear model predictive control via orthogonal search tree. *IEEE Transactions on Automatic Control*, *48*, 810–815.

Kvasnica, M., Christophersen, F.J., Herceg, M., & Fikar, M. (2008). Polynomial approximation of closed-form MPC for piecewise affine systems. In *Proceedings of the 17th IFAC world congress. Seoul, Korea* (pp. 3877–3882).

Kvasnica, M., & Fikar, M. (2010). Performance-lossless complexity reduction in explicit MPC. In *Conference on decision and control. CDC. Atlanta, USA* (pp. 5270–5275).

Kvasnica, M., Grieder, P., & Baotić, M. (2004). Multi-parametric toolbox (MPT). Available from: http://control.ee.ethz.ch/~mpt/.

Kvasnica, M., Löfberg, J., Herceg, M., Čirka, L'., & Fikar, M. (2010). Low-complexity polynomial approximation of explicit MPC via linear programming. In *Proceedings of the American control conference. Baltimore, USA* (pp. 4713–4718).

Lazar, M., de la Pena, D. Munoz, Heemels, W. P. M. H., & Alamo, T. (2008). On input-to-state stability of min–max nonlinear model predictive control. *Systems & Control Letters*, *57*, 39–48.

Löfberg, J. (2004). YALMIP: a toolbox for modeling and optimization in MATLAB. In *Proc. of the CACSD conference. Taipei, Taiwan*. Available from: http://users.isy.liu.se/johanl/yalmip/.

Mok, H., & To, W. (2008). Effective Pólya semi-positivity for non-negative polynomials on the simplex. *Journal of Complexity*, *24*(4), 524–544.

Tøndel, P., & Johansen, T.A. (2002). Complexity reduction in explicit linear model predictive control. In *Proc. of 15-th IFAC world congress*.

Tøndel, P., Johansen, T. A., & Bemporad, A. (2003). Evaluation of piecewise affine control via binary search tree. *Automatica*, *39*(5), 945–950.

Wen, Ch., Ma, X., & Ydstie, B. E. (2009). Analytical expression of explicit MPC solution via lattice piecewise-affine function. *Automatica*, *45*(4), 910–917.

Michal Kvasnica was born in 1977. He received his diploma in chemical engineering from the Slovak University of Technology in Bratislava, Slovakia and the Ph.D. in electrical engineering from the Swiss Federal Institute of Technology in Zurich, Switzerland. Since 2006 he is an Assistant Professor at the Slovak University of Technology in Bratislava. His research interests are in model predictive control, modeling of hybrid systems, and development of software tools for control. He is the co-author and developer of the MPT Toolbox for explicit model predictive control.

Johan Löfberg was born in Jönköping, Sweden in 1974. He received his M.Sc. degree in Mechanical Engineering in 1998 and the Ph.D. degree in Automatic Control in 2003, both from Linköping University, Linköping, Sweden. He held a post-doctoral position at the Automatic Control Laboratory at ETH Zurich, Switzerland 2003–2006. From 2006, he works as an Assistant Professor at Linköping University. His research interests include optimal and robust control in a model predictive control framework, and general aspects related to the use of optimization in control. He is the author and developer of the modeling language YALMIP.

Miroslav Fikar was born in Bratislava, Slovakia, in 1966 and received his ME and Ph.D. degrees in Chemical Engineering from the Slovak University of Technology in Bratislava in 1989 and 1994, respectively. He has stayed with the Faculty of Chemical and Food Technology STU where he is currently a professor and institute director. He was Postdoc Fellow in Nancy, France (Elf Aquitaine, Ministére de la Recherche), Alexander von Humboldt Fellow in Bochum, Germany and has spent several stays in Denmark, Germany, France, Switzerland. His current research interests include optimal control, MPC, and chemical process control.