Nearly-Optimal Simple Explicit MPC Regulators with Recursive Feasibility Guarantees

Takacs, B.; Holaza, J.; Kvasnica, M.; Di Cairano, S.

TR2013-101 December 2013

Abstract

Explicit Model Predictive Control (MPC) is an attractive control strategy, especially when one aims at a fast, computationally less demanding implementation of MPC. Although leading to a fast implementation of optimization based control, the main downside of explicit MPC is its high complexity in terms of memory occupancy, which often limits practical applicability of such a control methodology. Therefore in this paper we propose to obtain simple explicit MPC controllers that provide guarantees of recursive satisfaction of input and state constraints. The task is accomplished by optimizing, off-line, the parameters of the feedback law such that an integrated square error between the optimal, but complex controller and its simpler replacement is minimized. We show that the task can be formulated as a quadratic optimization problem which always yields an admissible solution. In this way, suboptimality of simple feedbacks with respect to their complex optimal counterparts is significantly mitigated.

IEEE Conference on Decision and Control (CDC)

Nearly-Optimal Simple Explicit MPC Regulators with Recursive Feasibility Guarantees

Bálint Takács, Juraj Holaza, Michal Kvasnica, and Stefano Di Cairano

***Abstract*— Explicit Model Predictive Control (MPC) is an attractive control strategy, especially when one aims at a fast, computationally less demanding implementation of MPC. Although leading to a fast implementation of optimization-based control, the main downside of explicit MPC is its high complexity in terms of memory occupancy, which often limits practical applicability of such a control methodology. Therefore in this paper we propose to obtain simple explicit MPC controllers that provide guarantees of recursive satisfaction of input and state constraints. The task is accomplished by optimizing, off-line, the parameters of the feedback law such that an integrated square error between the optimal, but complex controller and its simpler replacement is minimized. We show that the task can be formulated as a quadratic optimization problem which always yields an admissible solution. In this way, suboptimality of simple feedbacks with respect to their complex optimal counterparts is significantly mitigated.**

I. Introduction

Model predictive control (MPC) has become a very popular control strategy especially in process control [1], [2]. MPC is endorsed mainly due to its natural capability of designing feedback controllers for large multi-input-multi-output (MIMO) systems, while considering all of the systems physical constraints which are implicitly embedded in the optimization problem. Solution of such an optimization problem yields a sequence of predicted optimal control inputs, from which only the first one is applied to the system to achieve feedback.

To mitigate the required computational effort, explicit MPC [3] was introduced. In explicit MPC we pre-compute the function that generates the sequence of optimal control inputs for each admissible initial condition. The task of obtaining optimal control inputs then reduces to a mere function evaluation, which can be performed efficiently even with small computational resources. The downside is that such a pre-computed explicit solution consumes lots of memory. As was shown by numerous authors (see e.g. [4], [5]), complexity of explicit MPC solutions grows exponentially with the prediction horizon. To satisfy limits of implementation hardware, it is therefore important to keep complexity of explicit MPC controllers at an acceptable level.

Numerous procedures for simplifying explicit MPC controllers were proposed in the literature. For a comprehensive overview, the interested reader is referred to [6, Chapter 6] and the references therein. From more recent results we can mention construction of suboptimal controllers by exploiting freedom of Lyapunov functions by [7], simplification of feedbacks by using canonical PWA functions [8], and approximation of the optimal feedback by polynomials [9], [10]. In short, most of the simplification techniques focus at replacing the original complex controller by a simpler feedback law. The replacement must be done such that the simpler controller provides recursive satisfaction of constraints, but is allowed to sacrifice optimality in favor of achieving smaller complexity.

A simplification procedure along these lines is proposed in this paper. We assume that we are given a complex explicit MPC feedback law $\mu(x)$, encoded as a Piecewise Affine (PWA) function of the state measurements x. Our objective is to replace $\mu(\cdot)$ by a simpler PWA function $\widetilde{\mu}(\cdot)$ such that: a) $\widetilde{\mu}(x)$ generates a feasible sequence of control inputs for all admissible values of x; and b) the integrated square error between $\mu(\cdot)$ and $\widetilde{\mu}(\cdot)$ is minimized. By doing so we obtain a simpler explicit feedback law $\widetilde{\mu}(\cdot)$, which is safe (i.e., provides constraint satisfaction), and is nearly optimal.

Designing an appropriate approximate controller $\widetilde{\mu}(\cdot)$ first requires construction of polytopic regions over which $\widetilde{\mu}(\cdot)$ is defined, and then synthesizing local affine expressions in each of the regions. We propose to approach the first task by solving a simpler MPC optimization problem with a shorter prediction horizon. In this way, we obtain a simple feedback $\widehat{\mu}(\cdot)$ as a PWA function. We then refine local affine expressions of this function as to obtain the function $\widetilde{\mu}(\cdot)$ such that the error between $\mu(\cdot)$ and $\widetilde{\mu}(\cdot)$ is minimized. We show that if the MPC problem is formulated with an additional set of constraints (which employs a control invariant set), then the problem of finding $\widetilde{\mu}(\cdot)$ is always feasible. In other words, we can always refine $\widehat{\mu}(\cdot)$ as to obtain a better-performing explicit controller $\widetilde{\mu}(\cdot)$.

II. Preliminaries and Problem Definition

A. Notation and Definitions

We denote by $\mathbb{R}$, $\mathbb{R}^n$ and by $\mathbb{R}^{n\times m}$ the real numbers, n-dimensional real vectors, and $n \times m$ dimensional real matrices, respectively. Furthermore, $\mathbb{N}$ denotes the set of non-negative integers, and $\mathbb{N}_i^j$ the set of consecutive integers, i.e., $\mathbb{N}_i^j = \{i, \dots, j\}$, $i \leq j$. For a vector-valued function $f : \mathbb{R}^n \to \mathbb{R}^m$, $\mathrm{dom}(f)$ denotes its domain. For an arbitrary set $\mathcal{S}$, $\mathrm{int}(\mathcal{S})$ denotes its interior.

Definition 2.1 (Polytope): A polytope $\mathcal{P}$ is a convex, closed, and bounded set defined as the intersection of a finite number c of closed affine half-spaces $a_i^T x \leq b_i$,

B. Takács, J. Holaza, and M. Kvasnica are with the Slovak University of Technology in Bratislava, Slovakia, {balint.takacs, juraj.holaza,michal.kvasnica}@stuba.sk. S. Di Cairano is with Mitsubishi Electric Research Laboratories, Cambridge, MA, USA, dicairano@ieee.org.

$a_i \in \mathbb{R}^n$, $b_i \in \mathbb{R}$, $\forall i \in \mathbb{N}_1^c$. Each polytope can be compactly represented as

$$\mathcal{P} = \{x \in \mathbb{R}^n \mid Ax \leq b\}, \tag{1}$$

with $A \in \mathbb{R}^{c\times n}$, $b \in \mathbb{R}^c$.

Definition 2.2: Every polytope $\mathcal{P} \subset \mathbb{R}^n$ in (1) can be equivalently written as

$$\mathcal{P} = \{x \mid x = \sum_i \lambda_i v_i,\ 0 \leq \lambda_i \leq 1,\ \sum_i \lambda_i = 1\}, \tag{2}$$

where $v_i \in \mathbb{R}^n$, $\forall i \in \mathbb{N}_1^M$ are vertices of the polytope.

Definition 2.3 (Polytopic partition): The collection of polytopes $\{\mathcal{R}_i\}_{i=1}^M$ is called the partition of polytope $\mathcal{Q}$ if:

1) $\mathcal{Q} = \cup_i \mathcal{R}_i$,
2) $\mathrm{int}(\mathcal{R}_i) \cap \mathrm{int}(\mathcal{R}_j) = \emptyset$, $\forall i \neq j$.

We call each polytope of the collection a *region* of the partition.

Definition 2.4 (Polytopic PWA function): Vector-valued function $f : \Omega \to \mathbb{R}^m$ is called PWA over polytopes if

1) $\Omega \subset \mathbb{R}^n$ is a polytope,
2) there exist polytopes $\mathcal{R}_i$, $i = 1, \ldots, M$ such that $\{\mathcal{R}_i\}_{i=1}^M$ is the partition of Ω,
3) for each $i \in \mathbb{N}_1^M$ we have $f(x) = F_i x + g_i$, with $F_i \in \mathbb{R}^{m\times n}$, $g_i \in \mathbb{R}^m$.

Definition 2.5 (Maximum control invariant set): Let $x_{k+1} = Ax_k + Bu_k$ be a linear system that is subject to constraints $x \in \mathcal{X}$, $u \in \mathcal{U}$, $\mathcal{X} \subseteq \mathbb{R}^n$, $\mathcal{U} \subseteq \mathbb{R}^m$. Then the set

$$\begin{aligned}\mathcal{C}_\infty = \{x_0 \in \mathcal{X} \mid \forall k \in \mathbb{N} : \exists u_k \in \mathcal{U} \text{ s.t.}\\ x_{k+1} = Ax_k + Bu_k \in \mathcal{X}\}\end{aligned} \tag{3}$$

is called the maximum control invariant set.

Remark 2.6: Under mild assumptions, the set $\mathcal{C}_\infty$ in (3) is a polytope (see e.g. [11]) which can be computed for instance by the MPT Toolbox [12]. □

B. Explicit Model Predictive Control

We consider control of linear discrete-time systems in the state-space form

$$x(t+1) = Ax(t) + Bu(t), \tag{4}$$

with $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, (A, B) controllable. System in (4) is subject to state and input constraints

$$x(t) \in \mathcal{X},\ u(t) \in \mathcal{U}, \tag{5}$$

where $\mathcal{X} \subset \mathbb{R}^n$, $\mathcal{U} \subset \mathbb{R}^m$ are polytopes that contain the origin in their respective interiors. We are interested in obtaining a feedback law $\mu : \mathbb{R}^n \to \mathbb{R}^m$ such that $u(t) = \mu(x(t))$ drives all states of (4) to the origin while providing recursive satisfaction of state and input constraints, i.e., $\forall t \in \mathbb{N}$ we have $x(t) \in \mathcal{X}$, $u(t) \in \mathcal{U}$.

As shown for instance in [3], the feedback law $\mu(x)$ can be obtained by computing the explicit representation of the optimizer to the following optimization problem:

$$\mu = \arg\min \sum_{k=0}^{N-1} (x_{k+1}^T Q_x x_{k+1} + u_k^T Q_u u_k) \tag{6a}$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ \forall k \in \mathbb{N}_0^{N-1} \tag{6b}$$

$$u_k \in \mathcal{U},\ \forall k \in \mathbb{N}_0^{N-1} \tag{6c}$$

$$x_0 \in \mathcal{C}_\infty,\ x_1 \in \mathcal{C}_\infty, \tag{6d}$$

where x_k, u_k denote, respectively, predictions of the states and inputs at the time step $t+k$, initialized from $x_0 = x(t)$. Moreover, $N \in \mathbb{N}$ is the prediction horizon and $Q_x \succeq 0$, $Q_u \succ 0$ are weighting matrices of appropriate dimensions. In the receding horizon implementation of MPC, we are only interested in the first element of the optimal sequence of inputs $U_N^* = [u_0^{*T}, \ldots, u_{N-1}^{*}{}^T]^T$. Hence the receding horizon feedback law is given by

$$\mu(x) := [I_{m\times m}\ 0_{m\times m}\ \cdots\ 0_{m\times m}] U_N^*. \tag{7}$$

Remark 2.7: Note that constraint (6d) implies that if $\mathcal{C}_\infty$ is a control invariant set satisfying (3), then $x_k \in \mathcal{X}$ will be satisfied $\forall k \in \mathbb{N}_0^N$. □

By solving (6) using *parametric programming* (see [4], [13]), one obtains the explicit representation of the explicit MPC feedback law $\mu(\cdot)$ in (7) as a function of the initial condition $x_0 = x(t)$:

$$\mu(x_0) := \begin{cases} F_1 x_0 + g_1 & \text{if } x_0 \in \mathcal{R}_1 \\ \vdots & \\ F_M x_0 + g_M & \text{if } x_0 \in \mathcal{R}_M, \end{cases} \tag{8}$$

with $F_i \in \mathbb{R}^{m\times n}$ and $g_i \in \mathbb{R}^m$.

Theorem 2.8 ([14]): The function $\mu : \mathbb{R}^n \to \mathbb{R}^m$ in (8) is a polytopic PWA function (cf. Definition 2.4) where $\mathcal{R}_i \subset \mathbb{R}^n$ are polytopes $\forall i \in \mathbb{N}_1^M$, and M denotes the total number of polytopes. Moreover, the domain of $\mu(\cdot)$ is $\Omega = \cup_i \mathcal{R}_i$ where Ω is a polytope such that

$$\Omega = \{x_0 \mid \exists u_0, \ldots, u_{N-1} \text{ s.t. (6c)} - \text{(6d) holds}\} \tag{9}$$

is the set of all initial conditions for which problem (6) is feasible. Furthermore, $\{\mathcal{R}_i\}$ is the partition of Ω, cf. Definition 2.3. ■

C. Problem Statement

The main issue of explicit MPC is that complexity of explicit MPC feedback controllers $\mu(\cdot)$ in (8), expressed by the number of polytopes M, grows quickly with the prediction horizon N. The more polytopes constitute $\mu(\cdot)$, the more memory is required to store the function in the control hardware and the longer it takes to obtain value of the optimizer for a particular value of the state measurements. Therefore we want to replace $\mu(\cdot)$ by a similar, yet less complex, PWA feedback law $\widetilde{\mu}(\cdot)$ while preserving recursive satisfaction of constraints in (5). The price we are willing to pay for obtaining a simpler representation is suboptimality of $\widetilde{\mu}(\cdot)$ with respect to the optimal representation $\mu(\cdot)$.

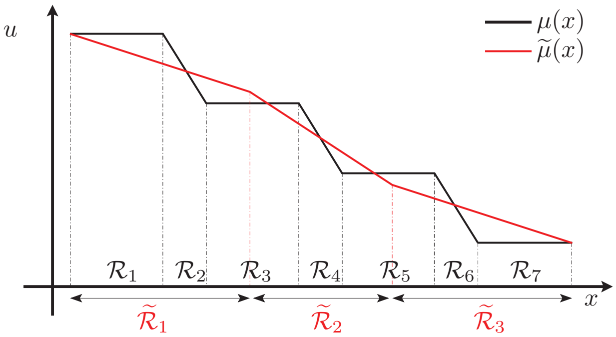


Fig. 1. The function $\mu(\cdot)$, shown in black, is given. The task in Problem 2.9 is to synthesize the function $\widetilde{\mu}(\cdot)$, shown in red, which is less complex (here it is defined just over 3 regions instead of 7 for $\mu(\cdot)$) and minimizes the integrated square error (11).

Problem 2.9: Given an explicit representation of the MPC controller $\mu : \mathbb{R}^n \to \mathbb{R}^m$ as in (8), we want to synthesize a PWA function $\widetilde{\mu} : \mathbb{R}^n \to \mathbb{R}^m$ with

$$\widetilde{\mu}(x) = \widetilde{F}_i x + \widetilde{g}_i \text{ if } x \in \widetilde{\mathcal{R}}_i,\ \forall i \in \mathbb{N}_1^{\widetilde{M}}, \tag{10}$$

i.e., to find the integer $\widetilde{M} < M$, polytopes $\widetilde{\mathcal{R}}_i \subset \mathbb{R}^n$, $i = 1, \ldots, \widetilde{M}$, and gains $\widetilde{F}_i \in \mathbb{R}^{m\times n}$, $\widetilde{g}_i \in \mathbb{R}^m$ such that:

- R1: for each $x \in \mathrm{dom}(\mu)$ we have that the simpler feedback $\widetilde{\mu}(\cdot)$ provides recursive satisfaction of state and input constraints in (5), i.e., $\forall t \in \mathbb{N}$ we have that $\widetilde{\mu}(x(t)) \in \mathcal{U}$ and $Ax(t) + B\widetilde{\mu}(x(t)) \in \mathcal{X}$;
- R2: $\widetilde{\mu}(\cdot)$ is chosen such that the squared error between the PWA functions $\mu(\cdot)$ and $\widetilde{\mu}(\cdot)$, when integrated over the domain of $\mu(\cdot)$, Ω, is minimized:

$$\min \int_{\Omega} \|\mu(x) - \widetilde{\mu}(x)\|_2^2 \, \mathrm{d}x. \tag{11}$$

□

In (11), $\mathrm{d}x$ is the Lebesgue measure of Ω, see [15]. The task of Problem 2.9 is illustrated graphically in Fig. 1.

Remark 2.10: It is important to note that solutions obtained by solving Problem 2.9 do not posses an a-priori guarantee of closed-loop stability. It is, however, straightforward to a-posteriori certify whether a particular feedback $\widetilde{\mu}(\cdot)$ possesses such a property by constructing a suitable Lyapunov function, see e.g. [16]. Moreover, in order to render the origin an equilibrium of system (4), the constraint $\widetilde{\mu}(0) = 0$ should be added to the set of requirements in Problem 2.9. □

III. Synthesis of Nearly-Optimal Simple Controllers

We propose to solve Problem 2.9 in two steps. In the first step, we construct the polytopes $\widetilde{\mathcal{R}}_i$, $i \in \mathbb{N}_1^{\widetilde{M}}$ with $\widetilde{M} < M$ (recall that M is the number of polytopes defining $\mu(\cdot)$) such that

$$\cup_i \widetilde{\mathcal{R}}_i = \cup_j \mathcal{R}_j, \tag{12}$$

hence such that the domain of $\widetilde{\mu}(\cdot)$ is identical to the domain of $\mu(\cdot)$. Then, in the second step, for each $i \in \mathbb{N}_1^{\widetilde{M}}$ we choose the gains $\widetilde{F}_i$ and offsets $\widetilde{g}_i$ in (10) such that $\widetilde{\mu}(\cdot)$ in (10) provides recursive satisfaction of constraints in (5) and the approximation error in (11) is minimized.

A. Selection of the Polytopic Partition

The objective here is to find polytopic regions $\widetilde{\mathcal{R}}_i$, $i \in \mathbb{N}_1^{\widetilde{M}}$ such that (12) holds with $\widetilde{M} < M$. First, recall that from Theorem 2.8, $\cup_j \mathcal{R}_j = \Omega$ by (9). Hence we require $\cup_i \widetilde{\mathcal{R}}_i = \Omega$. We propose to obtain polytopes $\widetilde{\mathcal{R}}_i$ by solving (6) again, but with a lower value of the prediction horizon, say with $\widehat{N} < N$, where N is the prediction horizon for which the original (complex) controller μ was obtained. Then, by Theorem 2.8, we obtain the feedback law $\widehat{\mu}(\cdot)$ as a PWA function of x:

$$\widehat{\mu}(x) = \widehat{F}_i x + \widehat{g}_i \text{ if } x \in \widetilde{\mathcal{R}}_i,\ \forall i \in \mathbb{N}_1^{\widetilde{M}}, \tag{13}$$

which is defined over $\widetilde{M}$ polytopes $\widetilde{\mathcal{R}}_i$.

Lemma 3.1: Let $\mu(\cdot)$ as in (8) be obtained by solving (6) according to Theorem 2.8 for some prediction horizon N. Let $\widehat{\mu}(\cdot)$ be the explicit MPC feedback function in (13), obtained by solving (6) for some $\widehat{N} < N$. Then (12) holds. ■

Hence we can obtain polytopic regions $\widetilde{\mathcal{R}}_i$ of the simpler function (10) by solving (6) explicitly for a shorter value of the prediction horizon. To achieve the least complex representation of $\widetilde{\mu}(\cdot)$, it is recommended to choose low values of $\widehat{N}$. The smallest number of polytopes, i.e., $\widetilde{M}$, will be achieved for $\widehat{N} = 1$.

Remark 3.2: The advantage of the procedure presented here is that the domain of $\mu(\cdot)$ is partitioned into $\{\widetilde{\mathcal{R}}_i\}$ in such a way that the approximation problem (11) is always feasible, i.e., there always exist parameters $\widetilde{F}_i$, $\widetilde{g}_i$ in (10) such that $\widetilde{\mu}$ guarantees recursive satisfaction of input and state constraints. This is not always the case if an arbitrary partition is selected. □

Remark 3.3: By solving (6) for $\widehat{N} < N$ we obtain the explicit representation of a simple controller $\widehat{\mu}(\cdot)$ as a PWA function in (13). Such a function already provides recursive satisfaction of constraints in (5) due to (6d), and therefore solves R1 in Problem 2.9. However, there is no guarantee that $\widehat{\mu}(\cdot)$ minimizes the approximation error (11). □

B. Function Fitting

In the previous section we have shown how to compute the polytopic partition $\{\widetilde{\mathcal{R}}_i\}$, $i = 1, \ldots, \widetilde{M}$ by solving (6) using parametric programming for $\widehat{N} < N$. Next we need to find parameters $\widetilde{F}_i$, $\widetilde{g}_i$ of $\widetilde{\mu}(\cdot)$ in (10) such that requirements R1 and R2 of Problem 2.9 are satisfied.

First, recall that, from Theorem 2.8, the polytopes $\widetilde{\mathcal{R}}_i$ form the partition of the domain of $\widehat{\mu}$, i.e., their respective interiors do not overlap. Therefore we can split the search for $\widetilde{F}_i$, $\widetilde{g}_i$ for $i = 1, \ldots, \widetilde{M}$ in Problem 2.9 into a series of $\widetilde{M}$ problems of the following form:

$$\min_{\widetilde{F}_i, \widetilde{g}_i} \int_{\widetilde{\mathcal{R}}_i} \|\mu(x) - \widetilde{\mu}(x)\|_2^2 \, \mathrm{d}x \tag{14a}$$

$$\text{s.t. } \forall x \in \widetilde{\mathcal{R}}_i : \begin{cases} \widetilde{F}_i x + \widetilde{g}_i \in \mathcal{U}, \\ Ax + B(\widetilde{F}_i x + \widetilde{g}_i) \in \mathcal{C}_\infty. \end{cases} \tag{14b}$$

Here, recall that $\widetilde{\mu}(x) = \widetilde{F}_i x + \widetilde{g}_i$ is the approximate affine control law valid for all $x \in \widetilde{\mathcal{R}}_i$ via (10). The first constraint in (14b) ensures satisfaction of input constraints, while the second case provides recursive satisfaction of state constraints since $\mathcal{C}_\infty$ is assumed to satisfy Definition 2.5.

However, there are three technical issues which complicate the search for $\widetilde{F}_i$, $\widetilde{g}_i$ from (14):

1) Even when x is restricted to a particular polytope $\widetilde{\mathcal{R}}_i$, $\mu(\cdot)$ over $\widetilde{\mathcal{R}}_i$ is still a PWA function, cf. Fig. 1.
2) The integration in (14a) has to be performed over polytopes in dimension $n \geq 1$.
3) The constraints in (14b) have to hold *for all* points $x \in \widetilde{\mathcal{R}}_i$, i.e., for an infinite number of points.

The first issue can be tackled as follows. Consider a fixed index i, i.e., take $\widetilde{\mathcal{R}}_i$, and recall that the (complex) optimal feedback $\mu(\cdot)$ is defined over M polytopes $\mathcal{R}_j$. For each $j = 1, \ldots, M$, compute first the intersection between $\widetilde{\mathcal{R}}_i$ and $\mathcal{R}_j$, i.e.,

$$\mathcal{Q}_{i,j} = \widetilde{\mathcal{R}}_i \cap \mathcal{R}_j, \ \forall j \in \mathbb{N}_1^M. \tag{15}$$

Indeed, each $\mathcal{Q}_{i,j}$ is a polytope. In each intersection $\mathcal{Q}_{i,j}$, the expressions for both $\mu(\cdot)$ and $\widetilde{\mu}(\cdot)$ are affine, which follows from (8). Hence, we can equivalently represent the approximation objective (14a) as

$$\min_{\widetilde{F}_i, \widetilde{g}_i} \sum_{j \in \mathcal{J}_i} \int_{\mathcal{Q}_{i,j}} \|(F_j x - g_j) - (\widetilde{F}_i x + \widetilde{g}_i)\|_2^2 \, \mathrm{d}x, \tag{16}$$

where F_j and g_j are the gains and offsets of the optimal feedback. The outer summation only needs to considers indices of polytopes of $\mu(\cdot)$ for which the intersection in (15) is non-empty, i.e., $\mathcal{J}_i = \{j \in \mathbb{N}_1^M \,|\, \widetilde{\mathcal{R}}_i \cap \mathcal{R}_j \neq \emptyset\}$ for a fixed i.

To evaluate the integral in (16) recall that, for each i-j combination, F_j and g_j are known matrices/vectors, but $\widetilde{F}_i$, $\widetilde{g}_i$ are optimized variables. Furthermore, $\mathcal{Q}_{i,j}$ are polytopes in $\mathbb{R}^n$. To obtain an analytic expression for the integral, we use the result of [17], extended by [15]:

Lemma 3.4 ([15]): Let f be a homogeneous polynomial of degree d in n variables, and let $s_1, \ldots, s_{n+1}$ be the vertices of an n-dimensional simplex Δ. Then

$$\int_\Delta f(y) \mathrm{d}y = \gamma \sum_{1 \leq i_1 \leq \cdots \leq i_d \leq n+1} \ \sum_{\epsilon \in \{\pm 1\}^d} \epsilon_1 \cdots \epsilon_d f(\textstyle\sum_{k=1}^d \epsilon_k s_{i_k}) \tag{17}$$

where

$$\gamma = \frac{\mathrm{vol}(\Delta)}{2^d d! \binom{d+n}{d}}, \tag{18}$$

and $\mathrm{vol}(\Delta)$ is the volume of the simplex. ■

However, Lemma 3.4 is not directly applicable to evaluate the integral in (16) because the polytopes $\mathcal{Q}_{i,j}$ are, in general, not simplices. To proceed, we therefore first have to tessellate each polytope $\mathcal{Q}_{i,j}$ into simplices $\Delta_{i,j,1}, \ldots, \Delta_{i,j,K}$ with $\mathrm{int}(\Delta_{i,j,k_1}) \cap \mathrm{int}(\Delta_{i,j,k_2}) = \emptyset$ for all $k_1 \neq k_2$, and $\cup_k \Delta_{i,j,k} = \mathcal{Q}_{i,j}$. Then we can rewrite (16) as a sum of the integrals evaluated over each simplex:

$$\min_{\widetilde{F}_i, \widetilde{g}_i} \sum_{j \in \mathcal{J}_i} \sum_{k=1}^K \int_{\Delta_{i,j,k}} \|(F_j x - g_j) - (\widetilde{F}_i x + \widetilde{g}_i)\|_2^2 \, \mathrm{d}x, \tag{19}$$

where K is the number of simplices tessellating $\mathcal{Q}_{i,j}$. Furthermore, note that Lemma 3.4 only applies to homogeneous polynomials. The integral error in (19), however, is not homogeneous. To see this, expand $f(x) := \|(F_j x + g_j) - (\widetilde{F}_i x + \widetilde{g}_i)\|_2^2$ to $f(x) := x^T Q x + r^T x + q$ with

$$Q = F_j^T F_j - 2F_j \widetilde{F}_i + \widetilde{F}_i^T \widetilde{F}_i, \tag{20a}$$

$$r = 2(F_j^T \widetilde{g}_i + \widetilde{F}_i^T \widetilde{g}_i - \widetilde{F}_i^T g_j - F_j^T \widetilde{g}_i), \tag{20b}$$

$$q = g_j^T g_j - 2g_j^T \widetilde{g}_i + \widetilde{g}_i^T \widetilde{g}_i. \tag{20c}$$

Then we can see that $f(x)$ is a quadratic function in the optimization variables $\widetilde{F}_i$ and $\widetilde{g}_i$, but is not homogeneous, since not all of its monomials have the same degree (in particular, we have monomials of degrees 2, 1 and 0 in f). However, since an integral is closed under linear combinations, we have that

$$\int_\Delta f(x) = \int_\Delta f_{\text{quad}}(x) + \int_\Delta f_{\text{lin}}(x) + \int_\Delta f_{\text{const}}, \tag{21}$$

with $f_{\text{quad}}(x) := x^T Q x$, $f_{\text{lin}} := r^T x$ and $f_{\text{const}} := q$ and the integrand $\mathrm{d}x$ is omitted for brevity. Since each of these newly defined functions is a homogeneous polynomial of degree 2, 1 and 0, respectively, the integral $\int_\Delta f(x) \mathrm{d}x$ can now be evaluated by applying (17) of Lemma 3.4 to each integral in the right-hand-side of (21). We hence obtain an analytic expression for the integral error as a quadratic function of the unknowns $\widetilde{F}_i$ and $\widetilde{g}_i$.

Remark 3.5: The integral of a constant q over a set $\Delta \subseteq \mathbb{R}^n$ is equal to a scaled volume of Δ, i.e., $\int_\Delta q = q \, \mathrm{vol}(\Delta)$. □

Finally, when optimizing for $\widetilde{F}_i$ and $\widetilde{g}_i$, we need to ensure that the constraints in (14b) hold for all points $x \in \widetilde{\mathcal{R}}_i$. By our assumptions, the sets $\mathcal{U}$ and $\mathcal{C}_\infty$ are polytopes, hence can be represented by $\mathcal{U} = \{u \,|\, H_{\mathrm{u}} u \leq h_{\mathrm{u}}\}$ and $\mathcal{C}_\infty = \{x \,|\, H_{\mathrm{c}} x \leq h_{\mathrm{c}}\}$. By using $u = \widetilde{F}_i x + \widetilde{g}_i$, constraint (14b) can be compactly written as

$$\forall x \in \widetilde{\mathcal{R}}_i : f(x) \leq 0, \tag{22}$$

with

$$f(x) := \begin{bmatrix} H_{\mathrm{u}} \widetilde{F}_i \\ H_{\mathrm{c}}(A + B\widetilde{F}_i) \end{bmatrix} x + \begin{bmatrix} H_{\mathrm{u}} \widetilde{g}_i - h_{\mathrm{u}} \\ H_{\mathrm{c}} \widetilde{g}_i - h_{\mathrm{c}} \end{bmatrix}. \tag{23}$$

Then we can state our next result.

Theorem 3.6: Let $\mathcal{V}_i = \{v_{i,1}, \ldots, v_{i,n_{v,i}}\}$, $v_{i,j} \in \mathbb{R}^n$ be the vertices of polytope $\widetilde{\mathcal{R}}_i$ (see Definition 2.2). Then (22) is satisfied $\forall x \in \widetilde{\mathcal{R}}_i$ if and only if $f(v_{i,j}) \leq 0$ holds for all vertices. ■

By combining Theorem 3.6 with the integration results reported above, we can hence equivalently recast the search for $\widetilde{F}_i$, $\widetilde{g}_i$ from (14) as

$$\min_{\widetilde{F}_i,\widetilde{g}_i} \sum_{j\in\mathcal{J}_i} \sum_{k=1}^{K} \int_{\Delta_{i,j,k}} \|(F_j x - g_j) - (\widetilde{F}_i x + \widetilde{g}_i)\|_2^2 \,\mathrm{d}x, \quad (24a)$$

$$\text{s.t. } \forall v_{i,\ell} \in \operatorname{vert}(\widetilde{\mathcal{R}}_i) : \begin{cases} \widetilde{F}_i v_{i,\ell} + \widetilde{g}_i \in \mathcal{U}, \\ A v_{i,\ell} + B(\widetilde{F}_i v_{i,\ell} + \widetilde{g}_i) \in \mathcal{C}_\infty, \end{cases} \quad (24b)$$

where $\operatorname{vert}(\widetilde{\mathcal{R}}_i)$ enumerates all vertices of the corresponding polytope. Since each polytope $\widetilde{\mathcal{R}}_i$ has only finitely many vertices [18], problem (24) has a finite number of constraints. Moreover, the objective in (24a) is a quadratic function in the unknowns $\widetilde{F}_i$, $\widetilde{g}_i$ and its analytic form can be obtained via (17). Finally, since the sets $\mathcal{U}$ and $\mathcal{C}_\infty$ are assumed to be polytopic, all constraints in (24b) are linear. Thus problem (24) is a quadratic optimization problem for each $i \in \mathbb{N}_1^{\widetilde{M}}$, where $\widetilde{M}$ is the number of polytopes that constitute the domain of $\widetilde{\mu}(\cdot)$ in (10).

As our final result we show that if polytopes $\widetilde{\mathcal{R}}_i$ are chosen as suggested by Lemma 3.1, then (24) is feasible for each $i = 1, \dots, \widetilde{M}$.

Theorem 3.7: Let $\widetilde{\mathcal{R}}_i$, $i = 1, \dots, \widetilde{M}$ be obtained by Lemma 3.1 for $\widehat{N} < N$. Then the optimization problem (24) is always feasible, i.e., for each $i = 1, \dots, \widetilde{M}$ there exist matrices $\widetilde{F}_i$ and vectors $\widetilde{g}_i$ such that the simplified feedback $\widetilde{\mu}(x)$ from (10) provides recursive satisfaction of constraints in (5) for an arbitrary $x \in \Omega$. ■

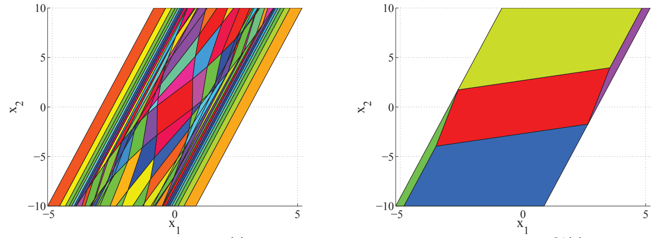
Remark 3.8: The improved feedback $\widetilde{\mu}(\cdot)$ in (10), whose parameters $\widetilde{F}_i$, $\widetilde{g}_i$ are obtained from (24), is not necessarily continuous. If desired, continuity can be enforced by adding the constraints $\widetilde{F}_i w_k + \widetilde{g}_i = \widetilde{F}_j w_k + \widetilde{g}_j$ to (24b), where w_k are all vertices of the $n-1$ dimensional intersection $\widetilde{\mathcal{R}}_i \cap \widetilde{\mathcal{R}}_j$, $\forall i, j \in \mathbb{N}_1^{\widetilde{M}}$. Note that, since the simple feedback $\widehat{\mu}$ is continuous, the choice $\widetilde{F}_i = \widehat{F}_i$, $\widetilde{g}_i = \widehat{g}_i$ is a feasible continuous solution in (24). Hence the conclusions of Theorem 3.6 hold even if continuity of (10) is enforced. Needless to say, sacrificing continuity allows for a greater reduction of the approximation error in (24a). □

Remark 3.9: The gain-optimization problem (24) naturally covers the multi-input scenario where $\widetilde{F}_i \in \mathbb{R}^{m\times n}$, $\widetilde{g}_i \in \mathbb{R}^m$ with $m > 1$. □

C. Complete Procedure

To solve Problem 2.9 and to devise a simpler explicit feedback law $\widetilde{\mu}(\cdot)$ in (10) that approximates a given complex solution $\mu(\cdot)$, we can proceed as follows:

1) Obtain $\widetilde{\mathcal{R}}_i$ by solving (6) for $\widehat{N} < N$, and for each $i \in \mathbb{N}_1^{\widetilde{M}}$ do:
2) Compute $\mathcal{Q}_{i,j}$ from (15) for each $j \in \mathbb{N}_1^M$.
3) Triangulate each intersection $\mathcal{Q}_{i,j}$ into simplices $\Delta_{i,j,1}, \dots, \Delta_{i,j,K}$ and enumerate their respective vertices.
4) Obtain the analytic expression of the integrals in (24a) by (17).
5) Enumerate vertices of $\widetilde{\mathcal{R}}_i$ and obtain $\widetilde{F}_i$, $\widetilde{g}_i$ by solving (24) as a quadratic optimization problem.



(a) 127 regions of $\mu(\cdot)$. (b) 5 regions of $\widetilde{\mu}(\cdot)$.

Fig. 2. Regions of the complex controller $\mu(\cdot)$ and of the approximate feedback $\widetilde{\mu}(\cdot)$ from Section **??**.

Obtaining the polytopes $\widetilde{\mathcal{R}}_i$ by solving (6) explicitly can be performed e.g. by the MPT Toolbox [12] or by the Hybrid Toolbox [19]. Computation of intersections, tessellation (via Delaunay triangulation), and enumeration of vertices in Steps 2 and 3 can also be done via MPT. Finally, the optimization problem (24) can be formulated by YALMIP [20] and solved using off-the-shelf software, e.g. by GUROBI [21] or `quadprog` of MATLAB.

IV. Example

Consider the second order, discrete-time, linear time-invariant system

$$x(t+1) = \begin{bmatrix} 0.9539 & -0.3440 \\ -0.4833 & -0.5325 \end{bmatrix} x(t) + \begin{bmatrix} -0.4817 \\ -0.5918 \end{bmatrix} u(t), \quad (25)$$

which is subject to state constraints $-10 \le x_i(t) \le 10$, $i \in \mathbb{N}_1^2$ and input bounds $-0.5 \le u(t) \le 0.5$. We remark that the system is open-loop unstable with eigenvalues $\lambda_1 = 1.0584$ and $\lambda_2 = -0.6370$. The complex explicit MPC controller $\mu(\cdot)$ in (8) was obtained by solving (6) for $Q_x = I_{2\times 2}$, $Q_u = 2$ and $N = 20$. Its explicit representation was defined over $M = 127$ polytopic regions $\mathcal{R}_i \subset \mathbb{R}^2$, shown in Fig. 2(a). All computations were carried out on a 2.7 GHz CPU using MATLAB and the MPT Toolbox.

To derive a simple representation of the MPC feedback as in (10), we have proceeded as outlined in Section III-C. First, we have solved (6) with shorter prediction horizons $\widehat{N} \in \{1, 2, 3, 4\}$. This gave rise to simple feedbacks $\widehat{\mu}(\cdot)$ as in (13) with lower performances. Domains of these feedbacks were defined, respectively, by $\widetilde{M} = \{3, 5, 11, 17\}$ regions $\widetilde{\mathcal{R}}_i$. These regions were then employed in (24) to optimize parameters $\widetilde{F}_i$, $\widetilde{g}_i$ of improved simple feedbacks $\widetilde{\mu}(\cdot)$ in (10). The fitting problems (24) were formulated by YALMIP and solved by `quadprog`.

Remark 4.1: In practice, to get the least complex approximate controller $\widetilde{\mu}(\cdot)$ one would only consider the case with the smallest number of regions. We only consider various values of $\widetilde{M}$ to assess suboptimality of $\widetilde{\mu}(\cdot)$ as a function of the number of regions, $\widetilde{M}$. □

Next, we have assessed degradation of performance induced by employing simpler feedbacks $\widehat{\mu}(\cdot)$ and $\widetilde{\mu}(\cdot)$ instead of the optimal controller $\mu(\cdot)$. To do so, for each suboptimal controller we have performed closed-loop simulations

TABLE I

COMPLEXITY AND SUBOPTIMALITY COMPARISON FOR THE EXAMPLE IN SECTION ??.

Pred. horizon	# of regions	Suboptimality w.r.t. $\mu(\cdot)$ in (8)	
		$\widehat{\mu}(\cdot)$ in (13)	$\widetilde{\mu}(\cdot)$ in (10)
1	3	60.8 %	25.1 %
2	5	32.9 %	18.0 %
3	11	11.4 %	8.3 %
4	17	6.9 %	1.7 %

for 10000 equidistantly spaced initial conditions from the domain of $\mu(\cdot)$. In each simulation we have evaluated the performance criterion $J_{\text{sim}} = \sum_{i=1}^{N_{\text{sim}}} x_i^T Q_x x_i + u_i^T Q_u u_i$ for $N_{\text{sim}} = 100$. For each investigated controller we have subsequently computed mean values of this criterion over all investigated starting points. This "average" performance indicators are denoted in the sequel as J_{opt} for the optimal feedback $\mu(\cdot)$, J_{simple} for the simple, but suboptimal controller $\widehat{\mu}(\cdot)$, and J_{improved} for $\widetilde{\mu}(\cdot)$, whose parameters were optimized in (24). Then we can express the average suboptimality of $\widetilde{\mu}(\cdot)$ by $J_{\text{simple}}/J_{\text{opt}}$, and the suboptimality of $\widetilde{\mu}(\cdot)$ by $J_{\text{improved}}/J_{\text{opt}}$. The higher the figure, the more suboptimal a respective controller is with respect to the optimal feedback $\mu(\cdot)$.

Concrete numbers are reported in Table I. As can be observed, lowering the prediction horizon significantly reduces complexity. However, suboptimality is inverse-proportional to complexity. For instance, solving (6) with $N = 1$ gives $\widehat{\mu}(\cdot)$ that performs by $60\,\%$ worse compared to the optimal feedback $\mu(\cdot)$ obtained for $N = 20$. Improving parameters of the feedback function via (24) resulted in an improved controller $\widetilde{\mu}(\cdot)$ whose average suboptimality is only $25\,\%$. The amount of suboptimality can be further reduced by considering more complex partition of the feedback function.

V. CONCLUSIONS

In this paper we have introduced a novel method for reducing complexity of explicit MPC controllers. The procedure was based on replacing regions of the complex feedback $\mu(\cdot)$ by a simpler partition $\{\widetilde{\mathcal{R}}_i\}$, followed by assigning to each region $\widetilde{\mathcal{R}}_i$ a local affine expression $\widetilde{F}_i x + \widetilde{g}_i$ such that loss of optimality with respect to μ is mitigated. The simpler partition was obtained by solving a simpler version of (6) with a lower value of the prediction horizon. Even though by doing so we already obtain a simpler feedback law $\widehat{\mu}(\cdot)$, by using the procedure of Section III-B we can significantly reduce the amount of suboptimality (cf. Remark 3.3). We have shown that the search for parameters $\widetilde{F}_i$, $\widetilde{g}_i$ in (10) can be formulated as a quadratic optimization problem. Moreover, we have proved that such a fitting problem is always feasible if a control invariant constraint is employed in (6d).

ACKNOWLEDGMENTS

B. Takács, J. Holaza and M. Kvasnica gratefully acknowledge the contribution of the Scientific Grant Agency of the Slovak Republic under the grant 1/0095/11, and the financial support of the Slovak Research and Development Agency under the project APVV 0551-11. This research was supported by Mitsubishi Electric Research Laboratories, under a Collaborative Research Agreement.

REFERENCES

[1] J. M. Maciejowski, *Predictive Control with Constraints*. Prentice Hall, 2002.

[2] S. J. Qin and T. A. Badgewell, "A survey of industrial model predictive control technology," vol. 11, pp. 733–764, 2003.

[3] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, Jan. 2002.

[4] F. Borrelli, *Constrained Optimal Control of Linear and Hybrid Systems*. Springer-Verlag, 2003, vol. 290.

[5] P. Grieder, "Efficient computation of feedback controllers for constrained systems," Ph.D. dissertation, Diss. ETH No. 15785, ETH Zürich, 2004.

[6] M. Kvasnica, *Real-Time Model Predictive Control via Multi-Parametric Programming: Theory and Tools*. Saarbruecken: VDM Verlag, Jan. 2009.

[7] L. Lu, W. Heemels, and A. Bemporad, "Synthesis of low-complexity stabilizing piecewise affine controllers: A control-Lyapunov function approach," in *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*. IEEE, 2011, pp. 1227–1232.

[8] A. Bemporad, A. Oliveri, T. Poggi, and M. Storace, "Ultra-fast stabilizing model predictive control via canonical piecewise affine approximations," *IEEE Trans. Automatic Control*, vol. 56, no. 12, pp. 2883–2897, 2011.

[9] G. Valencia-Palomo and J. Rossiter, "Using Laguerre functions to improve efciency of multi-parametric predictive control," in *Proceedings of the American Control Conference*, Baltimore, USA, 2010, pp. 4731–4736.

[10] M. Kvasnica, J. Löfberg, and M. Fikar, "Stabilizing polynomial approximation of explicit MPC," *Automatica*, vol. 47, no. 10, pp. 2292–2297, 2011.

[11] F. Blanchini and S. Miani, *Set-Theoretic Methods in Control*. Birkhauser Boston, 2008.

[12] M. Kvasnica, P. Grieder, and M. Baotić, "Multi-Parametric Toolbox (MPT)," 2004, available from http://control.ee.ethz.ch/~mpt/.

[13] M. Baotić, "Optimal Control of Piecewise Affine Systems – a Multi-parametric Approach," Dr. sc. thesis, ETH Zurich, Zurich, Switzerland, Mar. 2005.

[14] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," vol. 38, no. 1, pp. 3–20, Jan. 2002.

[15] V. Baldoni, N. Berline, J. A. De Loera, M. Köppe, and M. Vergne, "How to integrate a polynomial over a simplex," *Mathematics of Computation*, vol. 80, no. 273, p. 297, 2010.

[16] G. Ferrari-Trecate, F. A. Cuzzola, D. Mignone, and M. Morari, "Analysis of discrete-time piecewise affine and hybrid systems," *Automatica*, vol. 38, no. 12, pp. 2139–2146, Dec. 2002.

[17] J. Lasserre and K. Avrachenkov, "The Multi-Dimensional Version of $\int_a^b x^p dx$," *The American Mathematical Monthly*, vol. 108, no. 2, pp. 151–154, 2001.

[18] G. M. Ziegler, *Lectures on Polytopes*. Springer, 1994.

[19] A. Bemporad, *Hybrid Toolbox - User's Guide*, 2003. [Online]. Available: http://www.dii.unisi.it/hybrid/toolbox

[20] J. Löfberg, "YALMIP : A Toolbox for Modeling and Optimization in MATLAB," in *Proc. of the CACSD Conference*, Taipei, Taiwan, 2004, available from http://users.isy.liu.se/johanl/yalmip/.

[21] I. Gurobi Optimization, "Gurobi optimizer reference manual," 2012. [Online]. Available: http://www.gurobi.com