# Approximation of explicit model predictive control using regular piecewise affine functions : an input-to-state stability approach

## Document Version:

Accepted manuscript including changes made at the peer-review stage

## Please check the document version of this publication:

# Approximation of explicit model predictive control using regular piecewise affine functions: an input-to-state stability approach

B.A.G. Genuit, L. Lu and W.P.M.H. Heemels

## Abstract

Piecewise affine (PWA) feedback control laws defined on general polytopic partitions, as for instance obtained by explicit MPC, will often be prohibitively complex for fast systems. In this work we study the problem of approximating these high-complexity controllers by low-complexity PWA control laws defined on more regular partitions, facilitating faster on-line evaluation. The approach is based on the concept of input-to-state stability (ISS). In particular, the existence of an ISS Lyapunov function (LF) is exploited to obtain *a priori* conditions that guarantee asymptotic stability and constraint satisfaction of the approximate low-complexity controller. These conditions can be expressed as local semidefinite programs (SDPs) or linear programs (LPs), in case of 2-norm or $1, \infty$-norm based ISS, respectively, and apply to PWA plants. In addition, as ISS is a prerequisite for our approximation method, we provide two tractable computational methods for deriving the necessary ISS inequalities from nominal stability. A numerical example is provided that illustrates the main results.

# Approximation of explicit model predictive control using regular piecewise affine functions: an input-to-state stability approach

## I. INTRODUCTION

Piecewise affine (PWA) controllers have been a popular and powerful control solution for constrained linear and hybrid systems [1]–[5]. In many situations model predictive control (MPC), see e.g. the textbooks [6]–[8], has been used as a design methodology to obtain PWA control laws, as it indeed results under some conditions in an explicit PWA state feedback defined on a polytopic partition of the feasible set [9], [10]. Consequently, the on-line implementation of these so-called explicit MPC controllers is reduced from on-line optimization to a point location (a well-known problem in the field of geometrical computing, see e.g. [11]) on the polytopic partition and a corresponding lookup table with the PWA control parameters.

Unfortunately, point location on general polytopic partitions results in high on-line computational requirements, especially since the complexity (in terms of elementary operations and memory needed for the on-line evaluation) grows rapidly with the dimension of the state space. This is prohibitive for the implementation of these solutions on fast and/or large-scale applications, as controller evaluation times will rise above the admissible sampling period.

To overcome these limitations, research has been performed on efficient implementation of exact explicit MPC [12]–[14] and approximation algorithms to obtain low-complexity suboptimal controllers [15]–[22]. In particular, [12] presents a method to find a corresponding search tree for an existing partition to realize efficient implementations of the exact optimal PWA control law. This approach is further developed and implemented in [13]. In the research line based on approximation methods, [16] proposed an mp-QP approximation procedure imposing a hierarchical (also referred to as multiscale) hypercubic structure as the partition of the approximate PWA state feedback. This interesting method shares many properties with the one presented here, but applies to linear systems and quadratic costs, while our method in principle applies to PWA systems and both linear and quadratic costs. Based on convexity of the value function (optimal costs), [16] also provides *a priori* stability guarantees and performance bounds on the approximate control law. In [23] an interpolation-based method is presented to obtain approximate explicit MPC controllers for constrained linear systems using bilevel optimization techniques. A beneficial feature of the method is that it does not require to compute the optimal explicit MPC first, but still aims at minimizing the approximation error between the approximate and the optimal PWA control law. However, construction of the approximate explicit MPC law is based on an iterative procedure solving mixed integer linear programming (MILP) problems in each step, which can become computationally expensive in certain cases. Besides, no *a priori* stability guarantees are provided, only an *a posteriori* MILP-based stability test. The authors of [16] overcome the latter issues in [20], where an approximation method for constrained linear systems is proposed, which also uses a multiscale structure as in [16], but now based

on barycentric interpolation. Bounds guaranteeing stability and constraint satisfaction are obtained *a priori*. A lower bound on the performance is derived as well. This work presents an interesting approach that applies in the context of linear systems, exploiting convexity of the value function. As convexity and even continuity of the value function and control law might be lost for explicit MPC laws designed for PWA systems, this method and the one of [16] are not directly applicable to PWA systems. Also in [22] an approximation method for *linear* systems using PWA functions based on regular simplices is proposed, with guarantees of local optimality and constraint satisfaction. In addition, results of circuit implementation are presented. However, in [22] only *a posteriori* checks for stability are provided. In [19] an alternative method based on polynomial approximation of the optimal control law is presented. Here sum-of-squares (SOS) computations are used to compute the approximate polynomial control law that has *a priori* stability guarantees, constraint satisfaction and some degree of performance.

This paper proposes a novel approach to approximate (possibly discontinuous) PWA controllers with *a priori* guarantees on asymptotic stability and satisfaction of input and state constraints. The approximate control law is defined on a regular (possibly multiscale) partition, which can be chosen in a desirable way, e.g. based on multiscale rectangles as in [16] or regular simplices as in [20], [22], [24]–[27]. Due to the choice of a regular partition, the resulting regular PWA functions inherently result in efficient, low-complexity on-line implementations in terms of both computational effort and memory requirements. The proposed approach is based on the input-to-state stability (ISS) framework [28]–[30], computing *a priori* a lower bound on the robustness margin of the original high-complexity PWA controller against the approximation error. This bound is then used as a constraint for the approximation procedure to guarantee asymptotic stability (AS) of the plant in closed loop with the approximate low-complexity PWA controller. This constraint can be expressed as local semidefinite programs (SDPs) or (local) linear programs (LPs), depending if the ISS is based on 2-norms or $1, \infty$-norms, respectively. The main assets of our approach are flexibility (it can be used with any type of polytopic partition of the high- and low-complexity controllers), decoupled subproblems (facilitating parallel off-line computing and automated co-design of both the control parameters and the regular partition on which they are is defined), and its ability to handle discontinuous PWA controllers and plants. In addition, stability and constraint satisfaction are guaranteed *a priori* (as mentioned earlier) without needing convexity requirements on the value function corresponding to the original high-complexity controller as e.g. in [20]. These major advantages are obtained by requiring the original high-complexity controller to satisfy the ISS property, which under certain conditions is inherited from nominal stability of the high-complexity closed-loop system. Two tractable computational methods are provided in Section VI to show how and when nominal stability can be converted into the necessary ISS conditions. In other cases, e.g. for PWA plants not automatically resulting in ISS MPC controllers, the MPC setup has to be modified to guarantee the desired ISS property (see Remark 2 below). The latter approach is of particular interest as it might result in smaller ISS gains if compared to the inherent ISS gains. Smaller ISS gains typically result in a lower complexity of the resulting approximate PWA control laws.

The remainder of the paper is organized as follows. First, some notational conventions will be introduced to conclude this section. The necessary preliminaries will be discussed in Section II. The problem is outlined in Section III, and in Section IV, a central lemma is presented on which our approach is based. Section V discusses

the main approach and presents the computational algorithm. In Section VI two computational methods are described to obtain the ISS property from nominal stability. A numerical example for a PWA plant is presented in Section VII to illustrate our approach. Finally, conclusions are presented in Section VIII.

*A. Notations and basic definitions*

Let $\mathbb{R}$, $\mathbb{R}_+$, and $\mathbb{N}$ denote the set of real numbers, the set of non-negative reals, and the set of non-negative integers, respectively. We use the notation $\mathbb{N}_{\geq c_1}$ and $\mathbb{N}_{(c_1,c_2]}$ (*et similia*) to denote the sets $\{k \in \mathbb{N} \mid k \geq c_1\}$ and $\{k \in \mathbb{N} \mid c_1 < k \leq c_2\}$, for some $c_1, c_2 \in \mathbb{N}$. When inequalities such as $\leq$ (*et similia*) are applied to vectors, they should be interpreted elementwise throughout the paper. For matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n \times n}$, the inequality $\boldsymbol{A} \preceq \boldsymbol{B}$ (*et similia*) denotes that $\boldsymbol{A} - \boldsymbol{B}$ is a negative semidefinite matrix.

The Hölder $p$-norm of a vector $\boldsymbol{x} \in \mathbb{R}^{n_x}$ is defined as

$$\|\boldsymbol{x}\|_p = \begin{cases} \left(\sum_{n_x} |\boldsymbol{x}_i|^p\right)^{\frac{1}{p}}, & p \in \mathbb{N}_{[1,\infty)} \\ \max_i |\boldsymbol{x}_i|, & p = \infty, \end{cases} \tag{1}$$

where $\boldsymbol{x}_i$ is the $i$-th component of the vector $\boldsymbol{x}$, and $|\cdot|$ denotes the absolute value of a real number. When it is not important to specify the type of norm used explicitly, we just write $\|\cdot\|$. By $\boldsymbol{x}_i$ and $[\boldsymbol{x}]_i$, for $i \in \{1, \ldots, n_x\}$, we denote the $i$-th component of the vector $\boldsymbol{x} \in \mathbb{R}^{n_x}$. For a sequence $\{\boldsymbol{z}_p\}_{p \in \mathbb{N}}$ with $\boldsymbol{z}_p \in \mathbb{R}^l$, $p \in \mathbb{N}$, let $\|\{\boldsymbol{z}_p\}_{p \in \mathbb{N}}\| \triangleq \sup\{\|\boldsymbol{z}_p\| \mid p \in \mathbb{N}\}$ and let $\boldsymbol{z}_{[k]}$ denote the truncation of $\{\boldsymbol{z}_p\}_{p \in \mathbb{N}}$ at time $k \in \mathbb{N}$, i.e. $\boldsymbol{z}_{[k]} \triangleq \{\boldsymbol{z}_p\}_{p \in \mathbb{N}_{[0,k]}}$.

A function $\phi : \mathbb{R}_+ \to \mathbb{R}_+$ *belongs to class* $\mathcal{K}$ ($\phi \in \mathcal{K}$) if it is continuous, strictly increasing and $\phi(0) = 0$. A function $\phi : \mathbb{R}_+ \to \mathbb{R}_+$ *belongs to class* $\mathcal{K}_\infty$ ($\phi \in \mathcal{K}_\infty$) if $\phi \in \mathcal{K}$ and it is radially unbounded, i.e. $\lim_{s \to \infty} \phi(s) = \infty$. A function $\beta : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}_+$ *belongs to class* $\mathcal{KL}$ ($\beta \in \mathcal{KL}$) if for each fixed $t \in \mathbb{R}_+$, $\beta(\cdot, t) \in \mathcal{K}$ and for each fixed $s \in \mathbb{R}_+$, $\beta(s, \cdot)$ is non-increasing and $\lim_{t \to \infty} \beta(s, t) = 0$.

A set $\mathcal{P}$ is called a *polyhedron* if it can be written as the intersection of a finite number of half-spaces. A bounded polyhedron is called a *polytope*. A set of polytopes $\bar{\mathcal{P}} = \{\bar{\mathcal{P}}_1, \ldots, \bar{\mathcal{P}}_{n_{\bar{\mathcal{P}}}}\}$ is called a *partition of* $\mathbb{X}$ if $\cup_i \bar{\mathcal{P}}_i = \mathbb{X}$ and $\bar{\mathcal{P}}_i \cap \bar{\mathcal{P}}_j = \emptyset, \forall i, j \in \mathbb{N}_{\geq 1}, i \neq j$. The set of *extreme points* (or *vertices*) of a polytope $\mathcal{P}$ is denoted by $\text{ext}(\mathcal{P})$ and defined as the minimal set of points which convex hull equals the closure of the polytope $\mathcal{P}$ (denoted by $\text{cl}\,\mathcal{P}$). For a collection $\bar{\mathcal{P}} = \{\bar{\mathcal{P}}_1, \ldots, \bar{\mathcal{P}}_{n_{\bar{\mathcal{P}}}}\}$ of sets with $\bar{\mathcal{P}}_i \subseteq \mathbb{R}^{n_x}, i \in \{1, \ldots, n_{\bar{\mathcal{P}}}\}$, and another set $Q \subseteq \mathbb{R}^{n_x}$, the index set $\mathcal{I}(\mathcal{Q}, \bar{\mathcal{P}})$ is given by

$$\mathcal{I}(\mathcal{Q}, \bar{\mathcal{P}}) \triangleq \{\, i \in \{1, \ldots, n_{\bar{\mathcal{P}}}\} \mid \mathcal{Q} \cap \bar{\mathcal{P}}_i \neq \emptyset \,\}. \tag{2}$$

Hence, $\mathcal{I}(\mathcal{Q}, \bar{\mathcal{P}})$ is the set of all the indices of sets $\bar{\mathcal{P}}_i$ that have a non-empty intersection with $\mathcal{Q}$.

## II. PRELIMINARIES

Throughout this paper we will use the input-to-state stability (ISS) framework (see e.g. [28]) for discrete-time systems, see e.g. [30] and [31].

Consider a discrete-time dynamical system with state $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ and disturbance $\boldsymbol{e}_k \in \mathbb{R}^{n_e}$ at discrete time $k \in \mathbb{N}$, given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{e}_k), \tag{3}$$

where $\boldsymbol{g} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_e} \to \mathbb{R}^{n_x}$ is a nonlinear, possibly discontinuous function. We assume that the origin is an equilibrium of (3) in case of zero disturbance, i.e. $\boldsymbol{g}(0, 0) = 0$.

*Definition 1:* A set $\mathbb{P} \subseteq \mathbb{R}^{n_x}$ with $0 \in \text{int}(\mathbb{P})$ is called a *robustly positively invariant (RPI) set with respect to disturbance set* $\mathbb{E}$ for system (3) if for all $x \in \mathbb{P}$ and all $\boldsymbol{e} \in \mathbb{E}$ it holds that $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{e}) \in \mathbb{P}$ . In case system (3) does not depend on the disturbance $\boldsymbol{e}$, we call a set $\mathbb{P}$ that is RPI simply a *positively invariant (PI)* set. The following local notions of ISS and AS for discrete-time system (3) are used in this paper.

*Definition 2:* Let $\mathbb{X} \subseteq \mathbb{R}^{n_x}$ and $\mathbb{E} \subseteq \mathbb{R}^{n_e}$, with $0 \in \text{int}(\mathbb{X})$. We call system (3) *input-to-state stable (ISS) in* $\mathbb{X}$ *for disturbances in* $\mathbb{E}$ if there exist a $\mathcal{KL}$-function $\beta$ and a $\mathcal{K}$-function $\gamma$ such that, for each initial condition $\boldsymbol{x}_0 \in \mathbb{X}$ and error sequence $\{\boldsymbol{e}_p\}_{p \in \mathbb{N}}$, with $\boldsymbol{e}_p \in \mathbb{E}$ for all $p \in \mathbb{N}$, the corresponding state trajectory satisfies

$$\|\boldsymbol{x}_k\| \leq \beta(\|\boldsymbol{x}_0\|, k) + \gamma(\|\boldsymbol{e}_{[k-1]}\|), \forall k \in \mathbb{N}. \tag{4}$$

*Definition 3:* Let $\mathbb{X} \subseteq \mathbb{R}^{n_x}$, with $0 \in \text{int}(\mathbb{X})$. We call system (3) with $\boldsymbol{e} = 0$ *asymptotically stable (AS) in* $\mathbb{X}$ if there exists a $\mathcal{KL}$-function $\beta$ such that, for each initial condition $\boldsymbol{x}_0 \in \mathbb{X}$, the corresponding state trajectory satisfies

$$\|\boldsymbol{x}_k\| \leq \beta(\|\boldsymbol{x}_0\|, k), \forall k \in \mathbb{N}. \tag{5}$$

For determination of ISS and AS for discrete-time systems the following sufficient conditions can be used.

*Lemma 1:* [30], [31]: Let $\alpha_1, \alpha_2, \gamma, \sigma \in \mathcal{K}_\infty$. Let $\mathbb{X}$ with $0 \in \text{int}(\mathbb{X})$ be an RPI set with respect to $\mathbb{E}$ for system (3) and let $V : \mathbb{X} \to \mathbb{R}_+$ be a function with $V(0) = 0$. Consider the following inequalities:

$$\alpha_1(\|\boldsymbol{x}\|) \leq V(\boldsymbol{x}) \leq \alpha_2(\|\boldsymbol{x}\|), \tag{6a}$$

$$V(\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{e})) - V(\boldsymbol{x}) \leq -\gamma(\|\boldsymbol{x}\|) + \sigma(\|\boldsymbol{e}\|). \tag{6b}$$

If inequalities (6) hold for all $\boldsymbol{x} \in \mathbb{X}$ and all $\boldsymbol{e} \in \mathbb{E}$, then system (3) is ISS in $\mathbb{X}$ for disturbances in $\mathbb{E}$. If (6) holds for all $\boldsymbol{x} \in \mathbb{X}$ and $\boldsymbol{e} = 0$, and $\mathbb{X}$ is a PI set for (3) with zero disturbance, then the system (3) with $\boldsymbol{e} = 0$ is AS in $\mathbb{X}$.

A function $V$ that satisfies the hypotheses of Lemma 1 is called an *ISS Lyapunov function (ISS LF)* or a Lyapunov function (LF), respectively.

## III. PROBLEM STATEMENT

In this section we provide the problem statement and motivation for the studied problem.

### A. Setup

Consider a discrete-time dynamical system with input and state constraints, given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k) \tag{7a}$$

$$\boldsymbol{x}_k \in \mathbb{X} \triangleq \{\boldsymbol{x} \in \mathbb{R}^{n_x} \mid \boldsymbol{C_x x} \leq \boldsymbol{c_x}\} \tag{7b}$$

$$\boldsymbol{u}_k \in \mathbb{U} \triangleq \{\boldsymbol{u} \in \mathbb{R}^{n_u} \mid \boldsymbol{C_u u} \leq \boldsymbol{c_u}\}, \tag{7c}$$

where $k \in \mathbb{N}$ denotes the discrete time, and a PWA control law $\boldsymbol{u} : \mathbb{X}_f \to \mathbb{R}^{n_u}$ given by

$$\boldsymbol{u}(\boldsymbol{x}) = \begin{cases} \boldsymbol{F}_1 \boldsymbol{x} + \boldsymbol{g}_1, & \boldsymbol{x} \in \mathcal{P}_1 \\ \quad \vdots & \quad \vdots \\ \boldsymbol{F}_{n_{\mathcal{P}}} \boldsymbol{x} + \boldsymbol{g}_{n_{\mathcal{P}}}, & \boldsymbol{x} \in \mathcal{P}_{n_{\mathcal{P}}}. \end{cases} \tag{8}$$

Here, $\boldsymbol{C_x}$, $\boldsymbol{C_u}$, $\boldsymbol{F}_i$, and $\boldsymbol{c_x}$, $\boldsymbol{c_u}$, $\boldsymbol{g}_i$, $i \in \{1, \dots, n_{\mathcal{P}}\}$, are matrices and vectors of appropriate dimensions, respectively. We assume that $\mathcal{P}_i$ is polytopic and that

$$\text{cl } \mathcal{P}_i = \left\{ \boldsymbol{x} \in \mathbb{R}^{n_x} \left| \boldsymbol{C}_{\mathcal{P}_i} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix} \leq 0 \right. \right\}, i \in \{1, \dots, n_{\mathcal{P}}\}, \tag{9}$$

where $\boldsymbol{C}_{\mathcal{P}_i}$, $i \in \{1, \dots, n_{\mathcal{P}}\}$, are matrices of appropriate dimensions. Note that $\mathcal{P}_i$ is not necessarily a closed set. Due to this and the usage of non-strict inequalities in (9), cl $\mathcal{P}_i$ is expressed instead of $\mathcal{P}_i$ itself. In addition, we assume that

$$\mathcal{P} = \{\mathcal{P}_1, \dots, \mathcal{P}_{n_{\mathcal{P}}}\} \tag{10}$$

forms a partition of the feasible set $\mathbb{X}_f \subseteq \mathbb{X}$. The feasible set $\mathbb{X}_f$ is also assumed to be polytopic (and thus bounded) and given by

$$\mathbb{X}_f = \{\boldsymbol{x} \in \mathbb{R}^{n_x} \mid \boldsymbol{W}\boldsymbol{x} \leq \boldsymbol{w}\}, \tag{11}$$

for a matrix $\boldsymbol{W}$ and vector $\boldsymbol{w}$ of appropriate dimensions.

*Remark 1:* In this paper we focus on *hard* constraints, i.e. $\boldsymbol{x}_k \in \mathbb{X}$ should be satisfied for all $k \in \mathbb{N}$. This results in the feasible set on which the control law (12) is defined to be a subset of $\mathbb{X}$, i.e., $\mathbb{X}_f \subseteq \mathbb{X}$. However, if the state constraints $\boldsymbol{x}_k \in \mathbb{X}$ are interpreted as *soft* constraints, see e.g. [10], [22], the PWA law (12) might be defined on a feasible set $\mathbb{X}_f$ larger than $\mathbb{X}$. The techniques, which will be presented below, apply in a straightforward manner also in this setting.

*B. Motivation*

PWA functions as in (8) defined using general polytopic partitions often lead to high on-line computational and memory requirements, which are prohibitive for fast and/or large-scale applications, certainly in case of a high number of regions or state dimensions. This motivates the search for approximation methods for high-complexity PWA controllers leading to low-complexity controllers defined on partitions with less regions and/or regions of more *regular* shapes that enhance fast and memory-efficient on-line implementation, while still maintaining important closed-loop properties such as stability and constraint satisfaction.

As advocated in [16], [20], [22], [24]–[27], it can be very beneficial for the eventual (circuit) implementation to use canonical PWA controllers that are based on regular partitions using e.g. regular simplices [20], [22], [24]–[27] or (multiscale) hypercubes [16]. Essentially, any desired polytopic shape can be chosen in our approximation method that follows. For these reasons, we propose a method to approximate a given PWA state feedback law $\boldsymbol{u}$ as in (8) by a new control law $\tilde{\boldsymbol{u}} : \mathbb{X}_f \to \mathbb{R}^{n_u}$ given by

$$\tilde{\boldsymbol{u}}(\boldsymbol{x}) = \begin{cases} \tilde{\boldsymbol{F}}_1 \boldsymbol{x} + \tilde{\boldsymbol{g}}_1, & \boldsymbol{x} \in \tilde{\mathcal{P}}_1 \\ \quad \vdots & \quad \vdots \\ \tilde{\boldsymbol{F}}_{n_{\tilde{\mathcal{P}}}} \boldsymbol{x} + \tilde{\boldsymbol{g}}_{n_{\tilde{\mathcal{P}}}}, & \boldsymbol{x} \in \tilde{\mathcal{P}}_{n_{\tilde{\mathcal{P}}}}, \end{cases} \tag{12}$$

where the regions $\tilde{\mathcal{P}}_j$ have a more regular shape (e.g. simplicial or hypercubic, although the procedure allows any polytopic partition) and/or the number of regions in $\tilde{\mathcal{P}} = \left\{ \tilde{\mathcal{P}}_1, \ldots, \tilde{\mathcal{P}}_{n_{\tilde{\mathcal{P}}}} \right\}$ is smaller than the number of regions in the original partition $\mathcal{P}$, i.e. $n_{\tilde{\mathcal{P}}} < n_{\mathcal{P}}$. This new control law is required to asymptotically stabilize system (7) and satisfy state and input constraints (7b), (7c). In our setup we will allow the approximate control law (12) to be discontinuous, while, for instance, the works in [20], [22] use *continuous* PWA functions (defined on simplicial partitions) only.

Instrumental in the approach will be the study of the effect of the approximation error $\boldsymbol{e}_k = \tilde{\boldsymbol{u}}(\boldsymbol{x}_k) - \boldsymbol{u}(\boldsymbol{x}_k)$, $k \in \mathbb{N}$ on the closed-loop system. Therefore, we consider the perturbed closed-loop system

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}(\boldsymbol{x}_k) + \boldsymbol{e}_k), \tag{13}$$

and propose an approach that exploits ISS properties of the original (high-complexity) closed-loop system (13) with respect to $\boldsymbol{e}_k$. Note that the variable $\boldsymbol{e}_k$ can be interpreted as an actuator noise, but here it will play the role of the approximation error between the high-complexity and low-complexity controller. Such ISS properties are often inherited from nominal stability conditions for the closed-loop system, as will be discussed in detail in Section VI.

*Remark 2:* Alternatively to inherenting ISS from nominal stability, one can also directly design (explicit MPC) controllers that are ISS by combining ideas in [32] for constrained systems or [33] for unconstrained systems, together with the ISS MPC design techniques in e.g. [34]–[36]. This idea will be the objective of future research. Once an explicit PWA control law that is ISS with respect to actuator noise (approximation error) is obtained, or a nominally stabilizing PWA controller is available for which the conditions in Section VI hold, the techniques as presented in this paper are applicable.

This leads us to the following problem statement:

*Problem 1:* Given the constrained system (7), the PWA control law (8), with $\mathbb{X}_f \subseteq \mathbb{X}$, $0 \in \mathbb{X}_f$, and an ISS Lyapunov function $V : \mathbb{X}_f \to \mathbb{R}_+$ for the closed-loop system (13), find a PWA control law $\tilde{\boldsymbol{u}} : \mathbb{X}_f \to \mathbb{R}^{n_u}$ as in (12) approximating $\boldsymbol{u}$, defined on a more regular low-complexity partition $\tilde{\mathcal{P}}$ of $\mathbb{X}_f$, such that the resulting closed-loop system

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \tilde{\boldsymbol{u}}(\boldsymbol{x}_k)) \tag{14}$$

is asymptotically stable (AS) in $\mathbb{X}_f$, $\mathbb{X}_f$ is a PI set for the closed-loop system (14), and the input and state constraints (7b), (7c) are satisfied, i.e. $\mathbb{X}_f \subseteq \mathbb{X}$ and for all $\boldsymbol{x} \in \mathbb{X}_f$,

$$\tilde{\boldsymbol{u}}(\boldsymbol{x}) \in \mathbb{U}. \tag{15}$$

In the remainder of this paper, we will solve Problem 1 leading to a systematic procedure (with *a priori* conditions to guarantee stability and constraint satisfaction) to find a more regular control law $\tilde{\boldsymbol{u}}$ for (possibly discontinuous) constrained PWA plants (7).

*Remark 3:* For ease of exposition we assumed that $\tilde{\mathcal{P}} = \left\{ \tilde{\mathcal{P}}_1, \ldots, \tilde{\mathcal{P}}_{n_{\tilde{\mathcal{P}}}} \right\}$ is a partitioning of $\mathbb{X}_f$ in Problem 1. However, when applying the results in this paper in practice, one typically will choose a regular set $\mathcal{B}$ encompassing $\mathbb{X}_f$ in a tight manner, and define a partitioning $\tilde{\mathcal{P}}' = \left\{ \tilde{\mathcal{P}}'_1, \ldots, \tilde{\mathcal{P}}'_{n_{\tilde{\mathcal{P}}'}} \right\}$ of $\mathcal{B}$ instead, see Figure 1. Since $\mathcal{B}$ is regular (e.g. hypercubic), it can indeed be partitioned into smaller regular polytopes (e.g., hypercubes as used in [16] or regular simplices as adopted in [20], [22]), while this might be impossible for $\mathbb{X}_f$. However,
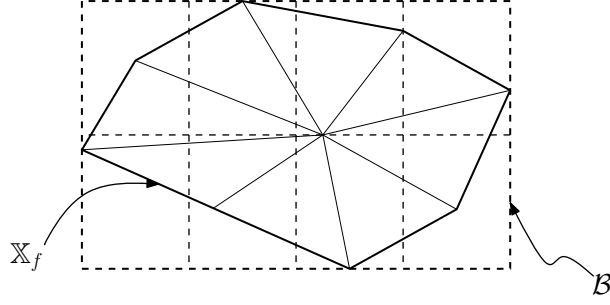
Fig. 1. Figure illustrating a possible choice for $\mathcal{B}$.

one can easily obtain a partitioning $\tilde{\mathcal{P}}$ of $\mathbb{X}_f$ from $\tilde{\mathcal{P}}'$ as defined in Problem 1 by taking $\tilde{\mathcal{P}}_i = \tilde{\mathcal{P}}'_i \cap \mathbb{X}_f$, for all $i = 1, 2, \ldots, n_{\tilde{\mathcal{P}}'}$. In fact, all properties regarding constraint satisfaction, stability and positive invariance will be guaranteed for $\mathbb{X}_f$ only, not for the complete set $\mathcal{B}$. However, in practice, one can still implement the control law defined on a partitioning $\tilde{\mathcal{P}}'$ of $\mathcal{B}$ and hence, the control law can be defined outside $\mathbb{X}_f$. This is of no concern as these parts will not be reached, because the computed approximate control law will guarantee that $\mathbb{X}_f$ is positively invariant and hence, whenever $\boldsymbol{x}_0 \in \mathbb{X}_f$, we have $\boldsymbol{x}_k \in \mathbb{X}_f$ for all $k \in \mathbb{N}$. As a consequence, even if the control law will be defined on $\mathcal{B}$ (and possibly outside $\mathbb{X}_f$), all provided properties are guaranteed when the initial condition $\boldsymbol{x}_0 \in \mathbb{X}_f$ holds.

## IV. A CENTRAL LEMMA

The following lemma will be instrumental in our developments. Starting from ISS of (13), and in particular the existence of an ISS LF, the lemma provides a bound on the approximation error $\|\tilde{\boldsymbol{u}}(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x})\|, \boldsymbol{x} \in \mathbb{X}_f$, that guarantees that the new control law $\tilde{\boldsymbol{u}}$ asymptotically stabilizes the original system (7a) in $\mathbb{X}_f$.

*Lemma 2:* Consider system (7) and suppose there exist a control law $\boldsymbol{u} : \mathbb{X}_f \to \mathbb{R}^{n_u}$ with $\mathbb{X}_f \subseteq \mathbb{X}$, $\alpha_1, \alpha_2, \gamma, \sigma \in \mathcal{K}_\infty$, a disturbance set $\mathbb{E}$ with $0 \in \mathbb{E}$ and a function $V : \mathbb{X}_f \to \mathbb{R}_+$ with $V(0) = 0$ such that for some $p, q \in \mathbb{N}_{[1,\infty)} \cup \{\infty\}$

$$\alpha_1(\|\boldsymbol{x}\|_q) \leq V(\boldsymbol{x}) \leq \alpha_2(\|\boldsymbol{x}\|_q), \tag{16a}$$

$$V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e})) - V(\boldsymbol{x}) \leq -\gamma(\|\boldsymbol{x}\|_q) + \sigma(\|\boldsymbol{e}\|_p), \tag{16b}$$

for all $\boldsymbol{x} \in \mathbb{X}_f$ and all $\boldsymbol{e} \in \mathbb{E}$. Then for any control law $\tilde{\boldsymbol{u}} : \mathbb{X}_f \to \mathbb{R}^{n_u}$ that satisfies

$$\sigma\left(\|\tilde{\boldsymbol{u}}(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x})\|_p\right) \leq \gamma\left(\|\boldsymbol{x}\|_q\right) - \tilde{\gamma}\left(\|\boldsymbol{x}\|_q\right), \tag{17}$$

$$\boldsymbol{f}(\boldsymbol{x}, \tilde{\boldsymbol{u}}(\boldsymbol{x})) \in \mathbb{X}_f, \tag{18}$$

and

$$\tilde{\boldsymbol{u}}(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x}) \in \mathbb{E}, \tag{19}$$

for some $\tilde{\gamma} \in \mathcal{K}_\infty$, and all $x \in \mathbb{X}_f$, the closed-loop system (14) is asymptotically stable in $\mathbb{X}_f$.

*Proof:* The proof follows from substitution of $\boldsymbol{e} = \tilde{\boldsymbol{u}}(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x})$, satisfying (19), in (16b) and by applying (17), which yields

$$V(\boldsymbol{f}(\boldsymbol{x}, \tilde{\boldsymbol{u}}(\boldsymbol{x}))) - V(\boldsymbol{x}) \leq -\tilde{\gamma}(\|\boldsymbol{x}\|_q), \forall \boldsymbol{x} \in \mathbb{X}_f \tag{20}$$

Equation (18) states that $\mathbb{X}_f$ is a PI set for system (14), which together with (20) and (16a) is sufficient for asymptotic stability in $\mathbb{X}_f$, according to Lemma 1. □

Using this lemma, we obtain that if $\tilde{\boldsymbol{u}}$ satisfies (15), (17), (18), and (19), then Problem 1 is solved, as due to (18) for any $\boldsymbol{x}_0 \in \mathbb{X}_f$, it also holds that $\boldsymbol{x}_k \in \mathbb{X}_f \subseteq \mathbb{X}$ for all $k \in \mathbb{N}$. Based on this central lemma, the question is now, given $\boldsymbol{u}$, how to construct such a control law $\tilde{\boldsymbol{u}}$ using computationally friendly tools.

A first step in this direction can be obtained by observing that if $\gamma \in \mathcal{K}_\infty$ and $\sigma \in \mathcal{K}_\infty$ have special forms, which is typically the case in explicit MPC, such as $\gamma(s) = \gamma_{\mathrm{c}} s^\mu$ and $\sigma(s) = \sigma_{\mathrm{c}} s^\mu$ for some $\gamma_{\mathrm{c}}, \sigma_{\mathrm{c}}, \mu \in \mathbb{R}_+$, and all $s \in \mathbb{R}_+$, then by selecting $\tilde{\gamma}(s) = \tilde{\gamma}_{\mathrm{c}} s^\mu$ with $0 < \tilde{\gamma}_{\mathrm{c}} < \gamma_{\mathrm{c}}$, (17) becomes

$$\|\tilde{\boldsymbol{u}}(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x})\|_p \leq \rho_{\max} \|\boldsymbol{x}\|_q \tag{21}$$

where

$$\rho_{\max} \triangleq \left( \frac{\gamma_{\mathrm{c}} - \tilde{\gamma}_{\mathrm{c}}}{\sigma_{\mathrm{c}}} \right)^{\frac{1}{\mu}}. \tag{22}$$

This bound on the approximation error can be calculated *a priori* and will be used to guarantee asymptotic stability for the approximate low-complexity controller $\tilde{\boldsymbol{u}}$.

In the following section we will apply this central lemma to the class of PWA systems and derive a systematic computational approach to find $\tilde{\boldsymbol{u}}$ satisfying (15), (18), (19), and (21) for all $\boldsymbol{x} \in \mathbb{X}_f$, thereby solving Problem 1.

## V. APPROACH FOR PWA SYSTEMS

Consider a (possibly discontinuous) PWA system given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}_r \boldsymbol{x}_k + \boldsymbol{B}_r \boldsymbol{u}_k + \boldsymbol{a}_r, \text{ when } \boldsymbol{x}_k \in \mathcal{S}_r \tag{23}$$

defined on a polytopic partition

$$\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_{n_\mathcal{S}}\}. \tag{24}$$

Suppose a PWA control law $\boldsymbol{u} : \mathbb{X}_f \to \mathbb{R}^{n_u}$ as in (8) with $\mathbb{X}_f \subseteq \cup_r \mathcal{S}_r$, and an ISS LF $V : \mathbb{X}_f \to \mathbb{R}_+$ exist, satisfying the conditions of Lemma 2 with functions $\gamma$ and $\sigma$ of the form as discussed at the end of Section III, leading to (21). To synthesize a regular approximate control law as in (12) we initially fix a new polytopic partition $\tilde{\mathcal{P}} = \left\{ \tilde{\mathcal{P}}_1, \ldots, \tilde{\mathcal{P}}_{n_{\tilde{\mathcal{P}}}} \right\}$ of $\mathbb{X}_f$ consisting of regular (e.g. hypercubic or regular simplicial) regions and show how the conditions (15), (18), (19), and (21) can be guaranteed locally for each $\tilde{\mathcal{P}}_j \in \tilde{\mathcal{P}}$, thereby solving Problem 1 for 23. In Section V-F, we will also provide an automated refinement procedure, which refines an initial coarse partition $\tilde{\mathcal{P}}_{\mathrm{init}}$ where necessary to satisfy the stability and constraint satisfaction requirements. Hence, although we start here with a fixed partition, a synthesis procedure for the regular partition will be provided as well.

### A. Asymptotic stability

To guarantee asymptotic stability, we will use (21) (next to positive invariance of $\mathbb{X}_f$ and $\mathbb{X}_f = \bigcup_j^{n_{\tilde{\mathcal{P}}}} \tilde{\mathcal{P}}_j$). Therefore, we write (21) for an arbitrary region $\tilde{\mathcal{P}}_j$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, in a computationally friendly form. For the Euclidean norm case ($p = q = 2$), we will make use of linear matrix inequalities (LMIs, [37]), for the linear norm case ($p, q \in \{1, \infty\}$), we will use linear programs (LPs).

*1) Euclidean norms* $(p = q = 2)$:

Since $\boldsymbol{u}$ and $\tilde{\boldsymbol{u}}$ are PWA functions as given in (8) and (12), respectively, (21) for all $\boldsymbol{x} \in \mathbb{X}_f$, can be written as

$$\| \underbrace{\begin{bmatrix} \tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i & \tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i \end{bmatrix}}_{\boldsymbol{H}_{ij}} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix} \|_2 \leq \rho_{\max} \|\boldsymbol{x}\|_2,$$

$$\forall \boldsymbol{x} \in \tilde{\mathcal{P}}_j \cap \mathcal{P}_i, \forall i \in \{1, \dots, n_{\mathcal{P}}\}, \tag{25}$$

$j \in \{1, \dots, n_{\tilde{\mathcal{P}}}\}$. To convert (25) into an LMI, we now take the matrix $\boldsymbol{E}_{ij}$ such that

$$\left\{ \boldsymbol{x} \in \mathbb{R}^{n_x} \,\middle|\, \boldsymbol{E}_{ij} \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix} \geq 0 \right\} = \mathrm{cl}\left( \tilde{\mathcal{P}}_j \cap \mathcal{P}_i \right). \tag{26}$$

Now note that (25) is implied (using the $\mathcal{S}$-procedure and Schur complements) by the LMI

$$\begin{bmatrix} \begin{bmatrix} \rho_{\max}^2 \boldsymbol{I}_{(n_x, n_x)} & 0 \\ 0 & 0 \end{bmatrix} - \boldsymbol{E}_{ij}^\top \boldsymbol{U}_{ij} \boldsymbol{E}_{ij} & \boldsymbol{H}_{ij}^\top \\ \boldsymbol{H}_{ij} & \boldsymbol{I}_{(n_u, n_u)} \end{bmatrix} \succeq 0,$$

$$\forall i \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{P}), \tag{27}$$

$j \in \{1, \dots, n_{\tilde{\mathcal{P}}}\}$, where $\boldsymbol{U}_{ij}$ is a symmetric matrix with nonnegative entries, and $\boldsymbol{H}_{ij} \triangleq \begin{bmatrix} \tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i & \tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i \end{bmatrix}$.

*Remark 4:* Note that setting up such LMIs is standard by now. In case of stability only, these are given in [38] and in [39], while an alternative technique for implementing the $\mathcal{S}$-procedure in discrete time has been proposed in [40]. The extension to ISS follows in a straightforward way leading to (27).

The conditions (27) are in LMI-form in the variables $\tilde{\boldsymbol{F}}_j, \tilde{\boldsymbol{g}}_j$ of the low-complexity controller corresponding to region $\tilde{\mathcal{P}}_j$, and $\boldsymbol{U}_{ij}$. Together with (18) and (19), affirming (27) for all $\tilde{\mathcal{P}}_j, j \in \{1, \dots, n_{\tilde{\mathcal{P}}}\}$, will guarantee asymptotic stability in $\mathbb{X}_f$ for the approximate control law $\tilde{\boldsymbol{u}}$.

*2) Linear norms* $(p, q \in \{1, \infty\})$:

For brevity we assume here $p, q = \infty$ in (21), as the other cases with $p, q \in \{1, \infty\}$ can be derived analogously. Similar to the Euclidean case, (21) can be written as

$$\|(\tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i)\boldsymbol{x} + (\tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i)\|_\infty \leq \rho_{\max} \|\boldsymbol{x}\|_\infty,$$

$$\forall \boldsymbol{x} \in \tilde{\mathcal{P}}_j \cap \mathcal{P}_i, \forall i \in \{1, \dots, n_{\mathcal{P}}\}, \tag{28}$$

$j \in \{1, \dots, n_{\tilde{\mathcal{P}}}\}$. Now consider the following theorem, graphically illustrated in Figure 2, where we denote $\tilde{\mathcal{P}}_j \cap \mathcal{P}_i$ as $\Lambda_{i,j}$ for compactness.

*Theorem 1:* Consider a polytope $\Lambda_{i,j} \subset \mathbb{R}^{n_x}$. Define the polytopes $\Lambda_{i,j}^{0,l}$ and $\Lambda_{i,j}^{1,l}$, $l \in \{1, \dots, n_x\}$, as

$$\Lambda_{i,j}^{0,l} \triangleq \{\boldsymbol{x} \in \Lambda_{i,j} \mid \|\boldsymbol{x}\|_\infty = \boldsymbol{x}_l\}, \tag{29a}$$

$$\Lambda_{i,j}^{1,l} \triangleq \{\boldsymbol{x} \in \Lambda_{i,j} \mid \|\boldsymbol{x}\|_\infty = -\boldsymbol{x}_l\}, \tag{29b}$$
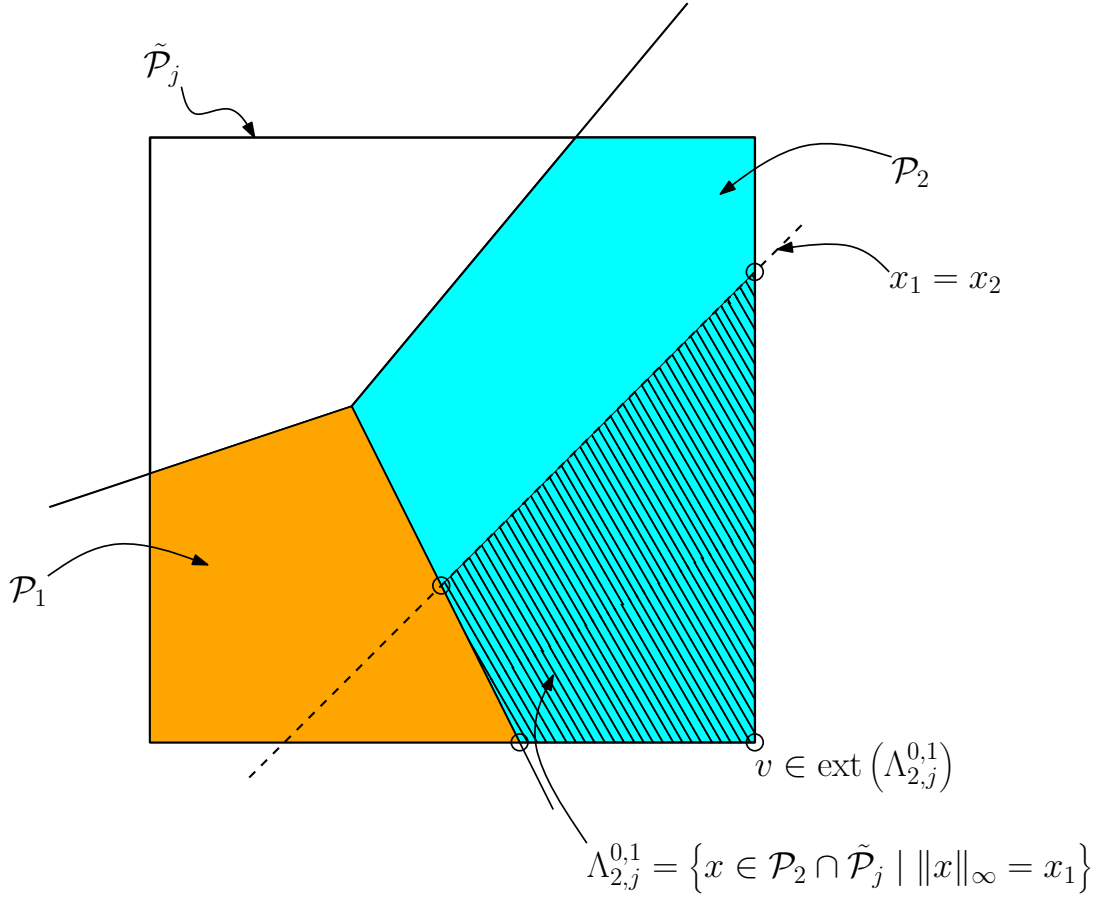
Fig. 2. Figure illustrating Theorem 1, where $\tilde{\mathcal{P}}_j$ is a box in the positive orthant of $\mathbb{R}^2$, $\Lambda_{2,j}^{0,1}$ is the shaded polytope and the vertices are indicated by circles ($\circ$).

where $\boldsymbol{x}_l$ denotes the $l$-th element of the vector $\boldsymbol{x}$. Then the following statements are equivalent:

(i) $\forall l \in \{1, \ldots, n_x\}, \forall s \in \{0, 1\}, \forall \boldsymbol{v} \in \text{ext}(\Lambda_{i,j}^{s,l})$,

$$\|(\tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i)\boldsymbol{v} + (\tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i)\|_\infty \leq (-1)^s \rho_{\max} [\boldsymbol{v}]_l. \tag{30}$$

(ii) $\forall \boldsymbol{x} \in \Lambda_{i,j}$,

$$\|(\tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i)\boldsymbol{x} + (\tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i)\|_\infty \leq \rho_{\max}\|\boldsymbol{x}\|_\infty. \tag{31}$$

*Proof:* To prove (i) $\Rightarrow$ (ii), observe that $\Lambda_{i,j} = \bigcup_{s=0}^1 \bigcup_{l=1}^{n_x} \Lambda_{i,j}^{s,l}$. As a consequence, it is sufficient to prove (31) for all $\boldsymbol{x} \in \Lambda_{i,j}^{s,l}$ and each $s \in \{0,1\}$ and $l \in \{1, \ldots, n_x\}$. Let $\boldsymbol{x} \in \Lambda_{i,j}^{s,l}$, then $\boldsymbol{x} = \sum_b \alpha_b \boldsymbol{v}^b$, for some $\boldsymbol{v}^1, \ldots, \boldsymbol{v}^{n_{s,l}} \in \text{ext}(\Lambda_{i,j}^{s,l})$ and some $\alpha_b \geq 0$, $b \in \{1, \ldots, n_{s,l}\}$, and $\sum_b \alpha_b = 1$. The left hand side of (31) for

some $\boldsymbol{x} \in \Lambda_{i,j}^{s,l}$ can be upperbounded using the triangle inequality and (30) as

$$
\begin{aligned}
&\|(\tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i)\boldsymbol{x} + (\tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i)\|_\infty \\
&= \| \sum_b \alpha_b(\tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i)\boldsymbol{v}^b + \sum_b \alpha_b(\tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i)\|_\infty \\
&\leq \sum_b \alpha_b\|(\tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i)\boldsymbol{v}^b + (\tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i)\|_\infty \\
&\overset{(30)}{\leq} \sum_b \alpha_b(-1)^s \rho_{\max} \left[\boldsymbol{v}^b\right]_l = (-1)^s \rho_{\max}\boldsymbol{x}_l, \\
&= \rho_{\max}\|\boldsymbol{x}\|_\infty.
\end{aligned}
\tag{32}
$$

In the last equality we used the definition of $\Lambda_{i,j}^{s,l}$ as in (29), implying that $\|\boldsymbol{x}\|_\infty = (-1)^s \boldsymbol{x}_l$ for $\boldsymbol{x} \in \Lambda_{i,j}^{s,l}$.

To show (ii) $\Rightarrow$ (i), note that for any $\boldsymbol{v} \in \mathrm{ext}(\Lambda_{i,j}^{s,l}) \subseteq \Lambda_{i,j}$, and $(-1)^s \left[\boldsymbol{v}\right]_l = \|\boldsymbol{v}\|_\infty$, due to the definition of $\Lambda_{i,j}^{s,l}$ as in (29). Hence, if (31) holds for all $\boldsymbol{x} \in \Lambda_{i,j}$, it also holds for $\boldsymbol{v}$ and thus we obtain (30). This completes the proof. $\qquad\square$

Hence, to compute $\tilde{\boldsymbol{F}}_j$ and $\tilde{\boldsymbol{g}}_j$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, satisfying (28), we apply Theorem 1 for each $\Lambda_{i,j} \triangleq \tilde{\mathcal{P}}_j \cap \mathcal{P}_i$, $i \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{P})$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$. Accordingly, (30) can be replaced by

$$
\pm \left[(\tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i)\boldsymbol{v} + (\tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i)\right]_{e_1} \leq (-1)^s \rho_{\max} \left[\boldsymbol{v}\right]_l,
$$
$$
\forall \boldsymbol{v} \in \mathrm{ext}(\Lambda_{i,j}^{s,l}), \forall s \in \{0, 1\}, \forall l \in \{1, \ldots, n_x\},
$$
$$
\forall e_1 \in \{1, \ldots, n_u\},
\tag{33}
$$

where $\Lambda_{i,j}^{s,l} \subseteq \Lambda_{i,j}$, $s \in \{0, 1\}$, $l \in \{1, \ldots, n_x\}$, denote the polytopes as defined in Theorem 1 using $\Lambda_{i,j} = \tilde{\mathcal{P}}_j \cap \mathcal{P}_i$, i.e.

$$
\Lambda_{i,j}^{s,l} = \left\{\boldsymbol{x} \in \tilde{\mathcal{P}}_j \cap \mathcal{P}_i \mid \|\boldsymbol{x}\|_\infty = (-1)^s \left[\boldsymbol{x}\right]_l\right\}.
\tag{34}
$$

Now, (33) for all $i \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{P})$ is an LP feasibility problem in $\tilde{\boldsymbol{F}}_j$ and $\tilde{\boldsymbol{g}}_j$ for region $\tilde{\mathcal{P}}_j$ to guarantee (21) (or (28) in this case) for all $\boldsymbol{x} \in \tilde{\mathcal{P}}_j$. Hence, to satisfy (28) (leading with (18) and (19) to asymptotic stability in $\mathbb{X}_f$), the LPs in (33) have to hold for all $i \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{P})$ and all $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$.

### B. Positive Invariance of $\mathbb{X}_f$

The desired invariance property (18) can be written for region $\tilde{\mathcal{P}}_j$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, as

$$
\boldsymbol{A}_r \boldsymbol{x} + \boldsymbol{B}_r \tilde{\boldsymbol{F}}_j \boldsymbol{x} + \boldsymbol{B}_r \tilde{\boldsymbol{g}}_j + \boldsymbol{a}_r \in \mathbb{X}_f,
$$
$$
\forall \boldsymbol{x} \in \tilde{\mathcal{P}}_j \cap \mathcal{S}_r, \forall r \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{S}).
\tag{35}
$$

Using convexity of $\tilde{\mathcal{P}}_j \cap \mathcal{S}_r$, and the explicit form of $\mathbb{X}_f$ as in (11), (35) can be written in terms of the vertices of each polytope $\tilde{\mathcal{P}}_j \cap \mathcal{S}_r$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, as

$$
\boldsymbol{W}\left(\boldsymbol{A}_r \boldsymbol{v} + \boldsymbol{B}_r \tilde{\boldsymbol{F}}_j \boldsymbol{v} + \boldsymbol{B}_r \tilde{\boldsymbol{g}}_j + \boldsymbol{a}_r\right) \leq \boldsymbol{w},
$$
$$
\forall \boldsymbol{v} \in \mathrm{ext}(\tilde{\mathcal{P}}_j \cap \mathcal{S}_r), \forall r \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{S}).
\tag{36}
$$

### C. Disturbance set $\mathbb{E}$ conditions

To guarantee the desired satisfaction of $e \in \mathbb{E}$ we assume $\mathbb{E}$ is polyhedral (with $0 \in \mathbb{E}$) and given by

$$\mathbb{E} = \left\{ \boldsymbol{e} \in \mathbb{R}^{n_u} \mid \boldsymbol{M}\boldsymbol{e} \leq \boldsymbol{m} \right\}, \tag{37}$$

for matrix $\boldsymbol{M}$ and vector $\boldsymbol{m}$ of appropriate dimensions. Hence, (19) is given for region $\tilde{\mathcal{P}}_j$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, as

$$\boldsymbol{M} \left( (\tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i)\boldsymbol{v} + (\tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i) \right) \leq \boldsymbol{m},$$

$$\forall \boldsymbol{v} \in \text{ext}(\tilde{\mathcal{P}}_j \cap \mathcal{P}_i), \forall i \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{P}). \tag{38}$$

### D. Input constraint satisfaction

Similarly, the input constraints (15) can be written for region $\tilde{\mathcal{P}}_j$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, as

$$\tilde{\boldsymbol{u}}\left(\boldsymbol{x}\right) \in \mathbb{U}, \forall \boldsymbol{x} \in \tilde{\mathcal{P}}_j \cap \mathbb{X}_f. \tag{39}$$

Again, using convexity of $\tilde{\mathcal{P}}_j \cap \mathbb{X}_f$, and the definition of $\mathbb{U}$ as in (7c), this can be written in terms of the vertices:

$$\boldsymbol{C_u} \left( \tilde{\boldsymbol{F}}_j \boldsymbol{v} + \tilde{\boldsymbol{g}}_j \right) \leq \boldsymbol{c_u}, \forall \boldsymbol{v} \in \text{ext}(\tilde{\mathcal{P}}_j \cap \mathbb{X}_f). \tag{40}$$

### E. Optimization problem

The result of Subsection V-A is either a semidefinite program (SDP) in case the 2-norm is used, or a linear program (LP) in case of $1, \infty$-norms. The results of Subsections V-B, V-C, and V-D are linear inequalities (36), (38), and (40) in the new control parameters, and can easily be added to the SDP or LP of Subsection V-A. The result is given for $\tilde{\mathcal{P}}_j$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, as

$$
\begin{aligned}
&\min_{\tilde{\boldsymbol{F}}_j, \tilde{\boldsymbol{g}}_j} \quad 0 \\
&\textbf{s.t.} \quad \textbf{a)} \text{ LMIs (27) or LP conds. (33)} \\
&\qquad \textbf{b)}\ \boldsymbol{W} \left( \boldsymbol{A}_r \boldsymbol{v} + \boldsymbol{B}_r \tilde{\boldsymbol{F}}_j \boldsymbol{v} + \boldsymbol{B}_r \tilde{\boldsymbol{g}}_j + \boldsymbol{a}_r \right) \leq \boldsymbol{w}, \\
&\qquad\quad \forall \boldsymbol{v} \in \text{ext}(\tilde{\mathcal{P}}_j \cap \mathcal{S}_r), \forall r \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{S}) \\
&\qquad \textbf{c)}\ \boldsymbol{M} \left( (\tilde{\boldsymbol{F}}_j - \boldsymbol{F}_i)\boldsymbol{v} + (\tilde{\boldsymbol{g}}_j - \boldsymbol{g}_i) \right) \leq \boldsymbol{m}, \\
&\qquad\quad \forall \boldsymbol{v} \in \text{ext}(\tilde{\mathcal{P}}_j \cap \mathcal{P}_i), \forall i \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{P}) \\
&\qquad \textbf{d)}\ \boldsymbol{C_u} \left( \tilde{\boldsymbol{F}}_j \boldsymbol{v} + \tilde{\boldsymbol{g}}_j \right) \leq \boldsymbol{c_u}, \\
&\qquad\quad \forall \boldsymbol{v} \in \text{ext}(\tilde{\mathcal{P}}_j \cap \mathbb{X}_f)
\end{aligned}
\tag{41}
$$

This convex optimization problem [41] can be solved by an SDP solver such as SeDuMi [42], or with an LP solver such as CPLEX [43] or GLPK [44], for the 2-norm and $1, \infty$-norm case, respectively. When a feasible solution to (41) is found for all $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, a control law $\tilde{\boldsymbol{u}}$ solving Problem 1 has been constructed.

Note that the problems for each $\tilde{\mathcal{P}}_j$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, are decoupled, which means (among others) that the total problem could be efficiently solved using parallel computing. Indeed, (41) has to be verified for each individual $\tilde{\mathcal{P}}_j$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, separately, instead of solving (41) as one monolithic problem for all $\tilde{\mathcal{P}}_j$

simultanously. In addition, the local character of the conditions allows that, in case the current region $\tilde{\mathcal{P}}_j$ results in an infeasible problem (41), the problem can again be solved for the refinement of $\tilde{\mathcal{P}}_j$ (i.e. $\tilde{\mathcal{P}}_j$ split in smaller regular subregions). A possible refinement procedure and corresponding algorithm are discussed in the next subsection.

*Remark 5:* Instead of only requiring feasibility in (41), one could also replace $\rho_{max}$ by $\rho_j$ in (27) or (33), and minimize $\max_j \rho_j$, while adding the additional constraint $0 < \rho_j \leq \rho_{max}$. This way, every local control law $\tilde{u}_j$ is not only feasible, but also as close as possible to the original (possibly optimal) $u$.

*F. Refinement procedure*

A possible refinement procedure can be obtained by exploiting the local character of the problems in (41). If, for some $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, (41) is not feasible, the region can be split into smaller regular subregions of which the local problem is given to the solver again.

*Remark 6:* This refinement implies a multiscale (or hierarchical) partition, which can be exploited to define a *search tree* [11] corresponding to the partition (e.g. a quadtree such as in Section VII), to further facilitate the efficient implementation of the point location, as already noted in [16]. Also in [12], [13] use is made of (binary) search trees for efficient point location.

This refinement procedure provides an enormous advantage in the sense that one can start from a very coarse initial partition $\tilde{\mathcal{P}}_{init}$ of $\mathbb{X}_f$ consisting of only a few regular regions and letting the automated refinement procedure determine where refinements are necessary in order to satisfy the stability and constraint satisfaction conditions. Hence, from this perspective one could perceive the refinement procedure as a method to synthesize, next to the control parameters $\tilde{F}_j$, $\tilde{g}_j$, $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, also the regular partition $\tilde{\mathcal{P}}$ itself.

To avoid that the refinement procedure does not end (keeps on splitting into subregions), a maximum number of regions $n_{max}$ and/or maximum level of refinement $h_{max}$ is added as a stopping criterium. A general setup for the refinement procedure including these options is presented in the next subsection.

*G. PWA Approximation Algorithm*

**Input:**    PWA control law $u$ and partition $\mathcal{P}$ as in (8),

            PWA plant dynamics and partition $\mathcal{S}$ as in (23),

            maximum 'approximation error' $\rho_{max}$ as in (22),

            maximum level depth $h_{max}$,

            maximal number of regions $n_{max}$,

            initial polytopic partition $\tilde{\mathcal{P}}_{init}$ of $\mathbb{X}_f$.

**Output:**  PWA control law $\tilde{u}$ and regular partition $\tilde{\mathcal{P}}$ as in (12), solving Problem 1.

1: initialize, $Bad := \tilde{\mathcal{P}}_{init}$, $Good := \emptyset$, $h(\mathcal{R}) := 1$ for all $\mathcal{R} \in \tilde{\mathcal{P}}_{init}$, $j := 0$

2: **while** $j \leq n_{max}$ **and** $Bad \neq \emptyset$ **do**

3:    select region $\mathcal{R}$ in $Bad$

4:    find the overlapping regions $\mathcal{I}(\mathcal{R}, \mathcal{P})$

5:    solve (41) for $\mathcal{R}$

6:    **if** (41) feasible **then**

7:  $\quad\quad Bad := Bad \setminus \{\mathcal{R}\}$

8:  $\quad\quad Good := Good \cup \{\mathcal{R}\}$

9:  $\quad\quad$ store control parameters $\tilde{\boldsymbol{F}}_j, \tilde{\boldsymbol{g}}_j$, corresponding to $\mathcal{R}$, obtained from (41)

10:  $\quad\quad j := j + 1$

11:  $\quad$ **else if** $h(\mathcal{R}) < h_{\max}$ **then**

12:  $\quad\quad$ split $\mathcal{R}$ in $L$ smaller regular subregions $\mathcal{R}_i$ s.t. $\{\mathcal{R}_1, \dots, \mathcal{R}_L\} = \mathcal{R}$

13:  $\quad\quad Bad := (Bad \setminus \{\mathcal{R}\}) \cup \{\mathcal{R}_1, \dots, \mathcal{R}_L\}$

14:  $\quad\quad$ store $h(\mathcal{R}_i) := h(\mathcal{R}) + 1, i = 1, \dots, L$

15:  $\quad$ **else**

16:  $\quad\quad$ output 'Warning: maximal level of refinement reached'

17:  $\quad\quad$ **end** (terminate algorithm)

18:  $\quad$ **end if**

19:  **end while**

20:  **if** $Bad = \emptyset$ **then**

21:  $\quad$ output 'Stabilizing approximation found'

22:  **else**

23:  $\quad$ output 'Warning: maximum number of regions reached'

24:  **end if**

25:  **end**

*Remark 7:* Several methods for splitting the regions (step 12 of the algorithm above) can be adopted. For instance, one could use binary refinement (also known as dyadic discretization) in case of a hypercubic partition to obtain $2^{n_x}$ smaller hypercubes (see Subsection VII for an illustration), multiscale regular simplicial as in [20] or in case of conic regions, see [45]. In particular, in case of a hypercubic partition and binary refinement, the result is an $n_x$–dimensional octree (generalized quadtree) (see e.g. [11]).

*Remark 8:* In case the refinement procedure in this section is infinite–recursively refining the regions around the origin (this might occur when the high-complexity control law $\boldsymbol{u}$ is not locally linear), it is recommended to relax the asymptotic stability requirement to an ultimate boundedness condition guaranteeing $\lim_{k \to \infty} \|\boldsymbol{x}_k\| \le \epsilon$ for a desirable size of the ultimate bound $\epsilon > 0$. This can be accomplished by relaxing (21), in case of e.g. $\mu = 1$, to

$$\|\tilde{\boldsymbol{u}}(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x})\|_p \le \rho_{\max}\|\boldsymbol{x}\|_q + \frac{\tilde{\gamma}_c \delta}{\sigma_c} \tag{42}$$

for some $\delta > 0$, which leads to a modification of (20) into

$$V(\boldsymbol{f}(\boldsymbol{x}, \tilde{\boldsymbol{u}}(\boldsymbol{x}))) - V(\boldsymbol{x}) \le -\tilde{\gamma}_c\|\boldsymbol{x}\|_q + \tilde{\gamma}_c\delta, \forall \boldsymbol{x} \in \mathbb{X}_f. \tag{43}$$

Inequality (43) can be used to guarantee ultimate boundedness to any arbitrarily small bound $\epsilon$, by appropriately selecting $\delta > 0$. The bound (42) provides a relaxation to (21) often avoiding infinite recursive refinement of the regions around the origin.

## VI. FROM NOMINAL STABILITY TO ISS

Many methods exist to design stabilizing MPC controllers for a system as in (7), including techniques based on terminal equality constraints, terminal set and costs, artificial Lyapunov functions, and so on, see e.g. [46] for an overview. By converting these MPC controllers into PWA state feedbacks using the explicit MPC techniques [9], [10], indeed stabilizing PWA controllers are obtained with an accompanying Lyapunov function (often being the value function of the MPC setup) proving the closed-loop stability. This Lyapunov function then satisfies inequalities as in (6a) and

$$V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}))) - V(\boldsymbol{x}) \leq -\gamma(\|\boldsymbol{x}\|) \tag{44}$$

for all $\boldsymbol{x} \in \mathbb{X}_f$, where $\gamma$ is a $\mathcal{K}_\infty$-function, and $\boldsymbol{u} : \mathbb{X}_f \to \mathbb{U}$ is the explicit optimal MPC law. It is now of interest to show under which conditions and how this nominal stability can be used to derive ISS using $V$ as an ISS Lyapunov function satisfying (16).

One such condition is the Lipschitz continuity of $V$, i.e., there exists a constant $L_V \geq 0$ such that for all $\boldsymbol{x}, \boldsymbol{z} \in \mathbb{X}_f$

$$|V(\boldsymbol{x}) - V(\boldsymbol{z})| \leq L_V \|\boldsymbol{x} - \boldsymbol{z}\|, \tag{45}$$

together with Lipschitz continuity of the dynamics $\boldsymbol{f}$ in the control input $\boldsymbol{u}$, i.e., there exists a constant $L_{\boldsymbol{f}} \geq 0$ such that for all $\boldsymbol{x} \in \mathbb{X}_f$ and all $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{U}$,

$$\|\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{v})\| \leq L_{\boldsymbol{f}} \|\boldsymbol{u} - \boldsymbol{v}\|, \tag{46}$$

equation (16) can be inherited from (44). Indeed, under these assumptions we have for all $\boldsymbol{x} \in \mathbb{X}_f$ and all $\boldsymbol{e}$ with $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e}) \in \mathbb{X}_f$ and $\boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e} \in \mathbb{U}$ that

$$
\begin{aligned}
&V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e})) - V(\boldsymbol{x}) \\
&= V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e})) - V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}))) \\
&\quad + V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}))) - V(\boldsymbol{x}) \\
&\leq L_V \|\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e}) - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}))\| - \gamma(\|\boldsymbol{x}\|) \\
&\leq -\gamma(\|\boldsymbol{x}\|) + L_V L_{\boldsymbol{f}} \|\boldsymbol{e}\|, \tag{47}
\end{aligned}
$$

which is an inequality of the type (16) and thus the ISS property of (13) with respect to $\boldsymbol{e}$ (with $\mathbb{E} = \mathbb{R}^{n_u}$) is proven. This would be one technique to obtain inherent ISS using global Lipschitz constants. Note that essentially what is needed to derive the above result is the existence of a $\sigma_c \geq 0$ such that

$$V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e})) - V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x})) \leq \sigma_c \|\boldsymbol{e}\| \tag{48}$$

for all $\boldsymbol{x} \in \mathbb{X}_f$ and all $\boldsymbol{e} \in \mathbb{R}^{n_u}$ with $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e}) \in \mathbb{X}_f$ and $\boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e} \in \mathbb{U}$. Actually, the bound in (48) might also be obtained for discontinuous $V$, see Remark 12 below.

*Remark 9:* To obtain ISS inherently in case of *discontinuous* systems and Lyapunov functions, use can be made of techniques presented in e.g. [47], which yields ISS with respect to additive disturbances. Using the method proposed in [32] and/or [33] these results can be converted in ISS with respect to $\boldsymbol{e}$ under certain conditions.

For two situations we will provide a more detailed computational approach leading to improved bounds in the sense of smaller $\sigma_\mathrm{c}$ in (48) and eventually larger $\rho_\mathrm{max}$ in (21):

1) PWA controllers as in (8) for PWA systems (23) (or linear systems) obtained from explicit LP-MPC, i.e. based on linear costs (using 1 and/or $\infty$-norms) using [48]. In this case the value function $V$ is a PWA function as well.

2) PWA controllers as in (8) for linear systems, obtained from explicit QP-MPC i.e. based on quadratic costs. In this case the value function $V$ is a continuous piecewise quadratic (PWQ) function, see [10].

To explain the two computational methods, recall (21) – (22), which will play an important role in the sequel. In particular, in the first case $\mu = 1$ and the norm is $\| \cdot \|_\infty$ (or $\| \cdot \|_1$) and in the second case $\mu = 2$ and the norm is $\| \cdot \|_2$.

### A. PWA value functions

When the MPC law $\boldsymbol{u} : \mathbb{X}_f \to \mathbb{U}$ was designed to be stabilizing for the PWA system (23), often the value function $V : \mathbb{X}_f \to \mathbb{R}_+$ is a Lyapunov function for the closed-loop dynamics $\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}(\boldsymbol{x}_k))$ and (44) holds. Let us now assume that $V$ is a *continuous* PWA function given by

$$V(\boldsymbol{x}) = \boldsymbol{H}_i \boldsymbol{x} + h_i, \boldsymbol{x} \in \mathcal{P}_i \tag{49}$$

for row vector $\boldsymbol{H}_i$ and scalar $h_i$, $i \in \{1, \ldots, n_\mathcal{P}\}$, and $\mathcal{P}$ is a polyhedral partition of $\mathbb{X}_f$. We will consider here the $\infty$-norm case, although for the 1-norm case the derivations are analogous.

*Remark 10:* For ease of exposition we assumed here that the partitions underlying $V$ and $\boldsymbol{u}$ are the same. If this is not the case, the calculations below can be adapted in a straightforward manner.

*1) Global Lipschitz bounds:* Similar to (47), in this case it is quite easy to see that

$$V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e})) - V(\boldsymbol{x}) \leq -\gamma_\mathrm{c} \|\boldsymbol{x}\|_\infty + \sigma_\mathrm{c} \|\boldsymbol{e}\|_\infty, \tag{50}$$

which is of the form (16b) (with $\| \cdot \|_\infty$ norms), when $\sigma_\mathrm{c}$ is taken to be

$$\sigma_\mathrm{c} = \max_{i=1,\ldots,n_\mathcal{P}} \max_{r=1,\ldots,n_\mathcal{S}} \|\boldsymbol{B}_r^\top \boldsymbol{H}_i^\top\|_1. \tag{51}$$

Note that $\|\boldsymbol{B}_r^\top \boldsymbol{H}_i^\top\|_1$ is the induced $\infty$-norm of $\boldsymbol{H}_i \boldsymbol{B}_r$. Moreover, $\mathbb{E} = \mathbb{R}^{n_u}$. Note that in proving (50) it is important to observe that due to the fact that the switching in (23) is only dependent on the state variable $\boldsymbol{x}$, the right-hand side of (23) is continuous in the control variable $\boldsymbol{u}$ for fixed $\boldsymbol{x}$, i.e. $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e}) = \boldsymbol{A}_r \boldsymbol{x} + \boldsymbol{B}_r \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{B}_r \boldsymbol{e} + \boldsymbol{a}_r$, when $\boldsymbol{x} \in \mathcal{S}_r$. Hence, to obtain (50) with $\sigma_\mathrm{c}$ as in (51) Lipschitz continuity of the value function as in (49) is indeed sufficient.

*2) Local Lipschitz bounds:* Reconsidering (33), which is a local problem (only related to the region $\tilde{\mathcal{P}}_j \cap \mathcal{P}_i$), one can imagine that it is possible to use more local versions of (21) with $\rho_\mathrm{max}$ becoming a function depending on the local region instead of using a global constant $\rho_\mathrm{max}$, which holds for all regions. Essentially, this idea is best perceived by replacing the constant $\sigma_\mathrm{c}$ by the piecewise constant function $\sigma_\mathrm{pc} : \mathbb{X}_f \to \mathbb{R}_+$ modifying (50) to

$$V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e})) - V(\boldsymbol{x}) \leq -\gamma_\mathrm{c} \|\boldsymbol{x}\|_\infty + \sigma_\mathrm{pc}(\boldsymbol{x}) \|\boldsymbol{e}\|_\infty \tag{52}$$

where now $\sigma_{\mathrm{pc}}$ satisfies $\sigma_{\mathrm{pc}}(\boldsymbol{x}) \leq \sigma_{\mathrm{c}}$ for some $\sigma_{\mathrm{c}} > 0$. In this way, locally the ISS constant $\sigma_{\mathrm{pc}}(\boldsymbol{x})$ can be smaller than the global ISS constant $\sigma_{\mathrm{c}}$ as in e.g. (50). This is clearly beneficial in the approximation problem. Indeed, (21) now becomes

$$\|\tilde{\boldsymbol{u}}(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x})\|_\infty \leq \rho_{\mathrm{max,pc}}(\boldsymbol{x})\|\boldsymbol{x}\|_\infty, \tag{53}$$

with

$$\rho_{\mathrm{max,pc}}(\boldsymbol{x}) = \frac{\gamma_{\mathrm{c}} - \tilde{\gamma}_{\mathrm{c}}}{\sigma_{\mathrm{pc}}(\boldsymbol{x})} \geq \frac{\gamma_{\mathrm{c}} - \tilde{\gamma}_{\mathrm{c}}}{\sigma_{\mathrm{c}}} = \rho_{\mathrm{max}}. \tag{54}$$

In this manner the approximation bound (21) becomes (53), which is less demanding and thus simpler PWA functions can be used to satisfy this bound. Indeed, larger values of $\rho_{\mathrm{max,pc}}(\boldsymbol{x})$ (than the worst-case global bound $\rho_{\mathrm{max}}$) allow for larger values of $\|\tilde{\boldsymbol{u}}(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x})\|$, thereby enhancing feasibility of (41).

Interestingly, this local Lipschitz constant $\sigma_{\mathrm{pc}}(\boldsymbol{x})$ can be computed using an LP when $V$ is a continuous PWA function. Consider the region $\Gamma_{i,j,r} \triangleq \tilde{\mathcal{P}}_j \cap \mathcal{P}_i \cap \mathcal{S}_r$ for some $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, $i \in \{1, \ldots, n_{\mathcal{P}}\}$ and $r \in \{1, \ldots, n_{\mathcal{S}}\}$. To bound $V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e})) - V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x})))$ for $\boldsymbol{x} \in \Gamma_{i,j,r}$ and $\boldsymbol{e} \in \mathbb{E}$, we compute the one-step reachable set from $\Gamma_{i,j,r}$,

$$R_{i,j,r} \triangleq \mathbb{X}_f \cap \left\{(\boldsymbol{A}_r + \boldsymbol{B}_r\boldsymbol{F}_j)\boldsymbol{x} + \boldsymbol{B}_r\boldsymbol{e} + \boldsymbol{B}_r\boldsymbol{g}_j + \boldsymbol{a}_r \mid \boldsymbol{x} \in \Gamma_{i,j,r}, \boldsymbol{F}_j\boldsymbol{x} + \boldsymbol{g}_j + \boldsymbol{e} \in \mathbb{U}, \boldsymbol{e} \in \mathbb{E}\right\}, \tag{55}$$

which is a polytopic set. Here $\mathbb{E}$ can be an arbitrarily chosen polytopic set with 0 in its interioir, providing a rough bound on the approximation error $\boldsymbol{e} = \tilde{\boldsymbol{u}}(\boldsymbol{x}) - \boldsymbol{u}(\boldsymbol{x})$, to restrict the search for the approximation law $\tilde{\boldsymbol{u}}$ and to further bound $R_{i,j,r}$. A local Lipschitz constant $\sigma_{\mathrm{c}}^{i,j,r}$ of $V$ on the set $R_{i,j,r}$ satisfying

$$V(\boldsymbol{x}) - V(\boldsymbol{z}) \leq \sigma_{\mathrm{c}}^{i,j,r}\|\boldsymbol{x} - \boldsymbol{z}\|_\infty \tag{56}$$

for all $\boldsymbol{x}, \boldsymbol{z} \in R_{i,j,r}$, can now be chosen according to

$$\sigma_{\mathrm{c}}^{i,j,r} = \max_{i \in \mathcal{I}(R_{i,j,r}, \mathcal{P})} \|\boldsymbol{B}_r^\top \boldsymbol{H}_i^\top\|_1. \tag{57}$$

Note that this choice complies with the piecewise constant function $\sigma_{\mathrm{pc}} : \mathbb{X}_f \to \mathbb{R}_+$ given by $\sigma_{\mathrm{pc}}(\boldsymbol{x}) = \sigma_{\mathrm{c}}^{i,j,r}$ for all $\boldsymbol{x} \in \Gamma_{i,j,r}$. Note that indeed $\sigma_{\mathrm{pc}}(\boldsymbol{x}) \leq \sigma_{\mathrm{c}}$ with $\sigma_{\mathrm{c}}$ as in (51). Hence, in the right-hand side of the inequality in (33) we can replace $\rho_{\mathrm{max}}$ by

$$\rho_{\mathrm{max}}^{i,j,r} = \frac{\gamma_{\mathrm{c}} - \tilde{\gamma}_{\mathrm{c}}}{\sigma_{\mathrm{c}}^{i,j,r}} \tag{58}$$

and carry out (33) for each set $\Gamma_{i,j,r}$ instead of $\Lambda_{i,j} = \tilde{\mathcal{P}}_j \cap \mathcal{P}_i$. Alternatively, if it is undesirable to carry out (33) for each $\Gamma_{i,j,r}$ but only for each $\Lambda_{i,j}$, one can replace $\rho_{\mathrm{max}}$ in (33) by

$$\rho_{\mathrm{max}}^{i,j} = \min_{r \in \mathcal{I}(\tilde{\mathcal{P}}_j \cap \mathcal{P}_i, \mathcal{S})} \frac{\gamma_{\mathrm{c}} - \tilde{\gamma}_{\mathrm{c}}}{\sigma_{\mathrm{c}}^{i,j,r}}, \tag{59}$$

which still yields better results than using the global bound in (51).

*Remark 11:* A further refinement is possible, leading to $\sigma_{\mathrm{pc}}(\boldsymbol{x})$ being smaller in some regions (at the cost of more off-line computations) by considering $\Lambda_{ij}^{s,l} \subset \tilde{\mathcal{P}}_j \cap \mathcal{P}_i$ and computing the constants $\sigma_{\mathrm{c}}^\bullet$ for each of these subsets $\Lambda_{i,j}^{s,l}$ of $\tilde{\mathcal{P}}_j \cap \mathcal{P}_i$ instead of for $\Lambda_{i,j} = \tilde{\mathcal{P}}_j \cap \mathcal{P}_i$ as above. This can be done similarly as outlined above by using $\Lambda_{i,j}^{s,l} \cap \mathcal{S}_r$ instead of $\mathcal{P}_i \cap \tilde{\mathcal{P}}_j \cap \mathcal{S}_r$.

*Remark 12:* Essentially this approach can also apply to discontinuous $V$ as long as $V$ is *continuous* on

$$\{\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e}) \in \mathbb{X}_f \mid \boldsymbol{e} \in \mathbb{E}, \boldsymbol{x} \in \mathbb{X}_f\}, \tag{60}$$

as then (56) still holds. In other words, if the one-step reachability set for the perturbed system does not intersect with discontinuities of $V$, this technique might still apply. See [47] for more details regarding this idea.

*B. PWQ value functions*

Let us consider a PWQ value function given by

$$V(\boldsymbol{x}) = V_i(\boldsymbol{x}) \triangleq \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}^\top \boldsymbol{H}_i \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}, \ x \in \mathcal{P}_i \tag{61}$$

for symmetric matrices

$$\boldsymbol{H}_i = \begin{bmatrix} \boldsymbol{H}_i^{11} & \boldsymbol{H}_i^{12} \\ \boldsymbol{H}_i^{21} & \boldsymbol{H}_i^{22} \end{bmatrix}, \ i = 1, \ldots, n_\mathcal{P}, \tag{62}$$

which are decomposed according to $\begin{bmatrix} \boldsymbol{x}^\top & 1 \end{bmatrix}^\top$, and $\mathcal{P}$ is a polytopic partition of $\mathbb{X}_f$. In the explicit MPC context this situation is studied in [10] for linear systems, i.e. $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{Ax} + \boldsymbol{Bu}$ for matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ of appropriate dimensions, and quadratic MPC costs.

Here we would like to provide an optimization-based method to obtain, for all $\boldsymbol{x} \in \mathbb{X}_f$ and all $\boldsymbol{e} \in \mathbb{E}$, with $\boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e} \in \mathbb{U}$,

$$V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e})) - V(\boldsymbol{x}) \leq -\gamma_\mathrm{c} \|\boldsymbol{x}\|_2^2 + \sigma_\mathrm{c} \|\boldsymbol{e}\|_2^2 \tag{63}$$

preferably with a minimal ratio $\frac{\sigma_\mathrm{c}}{\gamma_\mathrm{c}}$ (cf. (21) and (22)). This optimization can be obtained by first computing the set of all vectors $\begin{bmatrix} \boldsymbol{x}^\top & \boldsymbol{e}^\top \end{bmatrix}^\top$ for which the state $\boldsymbol{x} \in \mathcal{P}_i \cap \tilde{\mathcal{P}}_j$ is mapped to $\mathcal{P}_l$, i.e.,

$$\Theta_{i,j,l} = \left\{ \begin{bmatrix} \boldsymbol{x}^\top & \boldsymbol{e}^\top \end{bmatrix}^\top \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \middle| \boldsymbol{x} \in \mathcal{P}_i \cap \tilde{\mathcal{P}}_j, \right.$$

$$(\boldsymbol{A} + \boldsymbol{BF}_j)\boldsymbol{x} + \boldsymbol{Be} + \boldsymbol{Bg}_j \in \mathcal{P}_l, \boldsymbol{F}_j\boldsymbol{x} + \boldsymbol{g}_j + \boldsymbol{e} \in \mathbb{U},$$

$$\left. \boldsymbol{e} \in \mathbb{E} \right\}, \tag{64}$$

which is polyhedral provided $\mathbb{E}$ is. In this case we can write

$$\Theta_{i,j,l} = \left\{ \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{e} \end{bmatrix} \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \middle| \boldsymbol{M}_{i,j,l} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{e} \\ 1 \end{bmatrix} \geq 0 \right\} \tag{65}$$

for some matrix $\boldsymbol{M}_{i,j,l}$ of appropriate dimensions. Now (63) becomes

$$V_l((\boldsymbol{A} + \boldsymbol{BF}_i)\boldsymbol{x} + \boldsymbol{Bg}_i + \boldsymbol{Be}) - V_i(\boldsymbol{x}) \leq$$

$$- \gamma_\mathrm{c}^{i,j,l} \|\boldsymbol{x}\|_2^2 + \sigma_\mathrm{c}^{i,j,l} \|\boldsymbol{e}\|_2^2 \tag{66}$$

for all $\begin{bmatrix} \boldsymbol{x}^\top & \boldsymbol{e}^\top \end{bmatrix}^\top \in \Theta_{i,j,l}$, all $i \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{P})$, all $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, and all $l \in \{1, \ldots, n_\mathcal{P}\}$. Using the $\mathcal{S}$–procedure the quadratic constraint (66) is implied by the LMI (67) on page 19, where $\tilde{\boldsymbol{A}}_i = \boldsymbol{A} + \boldsymbol{BF}_i$, for all $i \in \mathcal{I}(\tilde{\mathcal{P}}_j, \mathcal{P})$, all $j \in \{1, \ldots, n_{\tilde{\mathcal{P}}}\}$, and all $l \in \{1, \ldots, n_\mathcal{P}\}$.

Obviously, (67) is an LMI in the parameters $\gamma_\mathrm{c}^{i,j,l} > 0$, $\sigma_\mathrm{c}^{i,j,l} \geq 0$ and the symmetric matrices $\boldsymbol{U}_{i,j,l}$ with nonnegative entries. Unfortunately, the objective function $\rho_\mathrm{max}^{i,j,l} = \frac{\gamma_\mathrm{c}^{i,j,l}}{\sigma_\mathrm{c}^{i,j,l}}$ that we try to maximize is not linear in these parameters. Therefore, one has to fix one of the parameters, say $\gamma_\mathrm{c}^{i,j,l}$ and then minimize $\sigma_\mathrm{c}^{i,j,l}$ given the

$$
\begin{bmatrix}
\tilde{\boldsymbol{A}}_i^\top \boldsymbol{H}_l^{11} \tilde{\boldsymbol{A}}_i - \boldsymbol{H}_i^{11} + \gamma_{\mathrm{c}}^{i,j,l} \boldsymbol{I}_{n_x,n_x} & \tilde{\boldsymbol{A}}_i^\top \boldsymbol{H}_l^{11} \boldsymbol{B} & \tilde{\boldsymbol{A}}_i^\top (\boldsymbol{H}_l^{11} \boldsymbol{B} \boldsymbol{g}_i + \boldsymbol{H}_l^{12}) - \boldsymbol{H}_i^{12} \\
\boldsymbol{B}^\top \boldsymbol{H}_l^{11} \tilde{\boldsymbol{A}}_i & \boldsymbol{B}^\top \boldsymbol{H}_l^{11} \boldsymbol{B} - \sigma_{\mathrm{c}}^{i,j,l} \boldsymbol{I}_{n_u,n_u} & \boldsymbol{B}^\top \boldsymbol{H}_l^{11} \boldsymbol{B} \boldsymbol{g}_i + \boldsymbol{B}^\top \boldsymbol{H}_l^{12} \\
((\boldsymbol{B}\boldsymbol{g}_i)^\top \boldsymbol{H}_l^{11} + \boldsymbol{H}_l^{21}) \tilde{\boldsymbol{A}}_i - \boldsymbol{H}_i^{21} & (\boldsymbol{B}\boldsymbol{g}_i)^\top \boldsymbol{H}_l^{11} \boldsymbol{B} + \boldsymbol{H}_i^{21} \boldsymbol{B} & \begin{bmatrix} \boldsymbol{B}\boldsymbol{g}_i \\ 1 \end{bmatrix}^\top \boldsymbol{H}_l \begin{bmatrix} \boldsymbol{B}\boldsymbol{g}_i \\ 1 \end{bmatrix} - \boldsymbol{H}_i^{22}
\end{bmatrix}
$$
$$
+ \boldsymbol{M}_{i,j,l}^\top \boldsymbol{U}_{i,j,l} \boldsymbol{M}_{i,j,l} \preceq 0, \tag{67}
$$

LMI constraint (67). With a line search in $\gamma_{\mathrm{c}}^{i,j,l}$ the optimal value for $\rho_{\max}^{i,j,l}$ can be found. This optimization has to be carried out for all regions $\Theta_{i,j,l}$ to get the global constants in (63) as

$$
\sigma_{\mathrm{c}} = \max_{i,j,l} \sigma_{\mathrm{c}}^{i,j,l} \text{ and } \gamma_{\mathrm{c}} = \min_{i,j,l} \gamma_{\mathrm{c}}^{i,j,l}. \tag{68}
$$

*Remark 13:* Instead of computing the global bounds as in (68), also here local values for $\gamma_{\mathrm{c}}^{i,j,l}$ and $\sigma_{\mathrm{c}}^{i,j,l}$ in (63) can be used by solving the optimization problems for restricted regions $\tilde{\mathcal{P}}_j \cap \mathcal{P}_i$ corresponding to (27). Hence, local approximation bounds are obtained as discussed in Subsection VI-A for the case of PWA value functions using local Lipschitz constants.

*Remark 14:* An optimization-based method as presented above for PWQ value functions can also be applied to PWA value functions resulting in LP problems.

*Remark 15:* The optimization-based approach above can also apply in the case of discontinuous Lyapunov functions $V$, provided finite values of $\sigma_{\mathrm{c}}$ and $\gamma_{\mathrm{c}}$ are found in (68).

## VII. EXAMPLE

We will present an example to illustrate our approach on a PWA plant for the $\infty$-norm case. In particular, we will approximate a stabilizing explicit PWA control law $\boldsymbol{u}$ obtained via explicit MPC, using the terminal cost and constraint set method [46] and computed using the multi-parametric toolbox (MPT, [49], version 2.6.2). To obtain an ISS LF, we will exploit Lipschitz continuity of the plant in $\boldsymbol{u}$ and of the value function, as also discussed in Section VI.

The example consists of the following PWA system, proposed in [4], with

$$
\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k) = \boldsymbol{A}_r \boldsymbol{x}_k + \boldsymbol{B}_r \boldsymbol{u}_k, \text{ when } \boldsymbol{x}_k \in \mathcal{S}_r, \text{ and } r = 1, 2, \tag{69}
$$

$$
\boldsymbol{A}_1 = \begin{bmatrix} 0.4 & 0.6928 \\ -0.6928 & 0.4 \end{bmatrix}, \boldsymbol{B}_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix},
$$
$$
\boldsymbol{A}_2 = \begin{bmatrix} 0.4 & -0.6928 \\ 0.6928 & 0.4 \end{bmatrix}, \boldsymbol{B}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{70}
$$

constraint sets

$$
\mathbb{U} = \left\{ \boldsymbol{u} \in \mathbb{R} \mid -1 \le \boldsymbol{u} \le 1 \right\}, \tag{71a}
$$

$$
\mathbb{X} = \left\{ \boldsymbol{x} \in \mathbb{R}^2 \mid -10 \le \boldsymbol{x} \le 10 \right\}, \tag{71b}
$$

and $\mathcal{S}_1 = \left\{ \boldsymbol{x} \in \mathbb{R}^2 \mid \begin{bmatrix} 1 & 0 \end{bmatrix} \boldsymbol{x} \geq 0 \right\}$, $\mathcal{S}_2 = \left\{ \boldsymbol{x} \in \mathbb{R}^2 \mid \begin{bmatrix} 1 & 0 \end{bmatrix} \boldsymbol{x} \leq 0 \right\}$. We append the standard MPC problem with a terminal cost and terminal set such that the cost function becomes

$$J(\boldsymbol{x}_k, \boldsymbol{u}_k) = \|\boldsymbol{P}\boldsymbol{x}_N\|_\infty + \sum_{l=1}^{N} \|\boldsymbol{Q}\boldsymbol{x}_l\|_\infty + \|\boldsymbol{R}\boldsymbol{u}_l\|_\infty, \tag{72}$$

with weighting matrices

$$\boldsymbol{Q} = \boldsymbol{I}, \tag{73}$$

$$\boldsymbol{R} = 1. \tag{74}$$

As said, we will use a terminal cost and set method [5], [46] and therefore will construct a matrix $\boldsymbol{P}$ and feedback gains $\boldsymbol{K}_r$, $r = 1, 2$, such that

$$\|\boldsymbol{P}(\boldsymbol{A}_r + \boldsymbol{B}_r\boldsymbol{K}_r)\boldsymbol{x}\|_\infty - \|\boldsymbol{P}\boldsymbol{x}\|_\infty + \|\boldsymbol{Q}\boldsymbol{x}\|_\infty$$
$$+ \|\boldsymbol{R}\boldsymbol{K}_r\boldsymbol{x}\|_\infty \leq 0, \tag{75}$$

for all $\boldsymbol{x} \in \mathbb{R}^{n_x}$ and $r = 1, 2$. This inequality implies (among others) that $\|\boldsymbol{P}\boldsymbol{x}\|_\infty$ is a Lyapunov function for the switched linear system given by

$$\boldsymbol{x}_{k+1} = (\boldsymbol{A}_r + \boldsymbol{B}_r\boldsymbol{K}_r)\boldsymbol{x}_k, r = 1, 2. \tag{76}$$

To find $\boldsymbol{P}$ and $\boldsymbol{K}_r$ satisfying (75), we use techniques presented in [5], resulting in

$$\boldsymbol{P} = \begin{bmatrix} 8.8933 & 0.0265 \\ 0.1588 & 14.2315 \end{bmatrix}. \tag{77}$$

Next, we take a level set of the Lyapunov function $\|\boldsymbol{P}\boldsymbol{x}\|_\infty$, which also satisfies input and state constraints for the auxiliary control law $\boldsymbol{u}_k = \boldsymbol{K}_r\boldsymbol{x}_k$, $\boldsymbol{u}_k \in \mathcal{S}_r$, $r = 1, 2$, as the terminal set $\mathbb{X}_N$. To be precise, we search for a $c_e \geq 0$ such that

$$\mathbb{X}_N = \{\boldsymbol{x} \in \mathbb{R}^{n_x} \mid \|\boldsymbol{P}\boldsymbol{x}\|_\infty \leq c_e\}, \tag{78a}$$

is inside the safe set $\{\boldsymbol{x} \in \mathbb{X} \mid \boldsymbol{K}\boldsymbol{x} \in \mathbb{U}\}$. In this case, $c_e = 13.3$ is the maximum number defining such a set.

Using the parameters above, and a prediction horizon $N = 7$, we compute the corresponding explicit MPC control law $\boldsymbol{u}$, shown in Figure 3, having 277 regions. Note that the controller can be simplified to 57 regions by merging regions containing the same control law. Using ideas in [12], the point location problem on this irregular partition can be represented by a binary search tree with a maximum depth of 9 levels, where a single boundary (i.e. a linear inequality) is evaluated at each node.

We will now apply the approach as discussed in Subsection VI-A using global Lipschitz continuity of the value function. The value function $V$ is a continuous PWA function and satisfies

$$V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})) - V(\boldsymbol{x}) \leq -\|\boldsymbol{Q}\boldsymbol{x}\|_\infty, \boldsymbol{x} \in \mathbb{X}_f \tag{79}$$

as follows from the basic proof of the terminal cost and set method [46]. Using the idead in Subsection VI-A we obtain for all $\boldsymbol{x} \in \mathbb{X}_f$ and $\boldsymbol{e} \in \mathbb{E}$ with $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x})) \in \mathbb{X}_f$

$$V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e}) - V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})) \leq \sigma_c \|\boldsymbol{e}\|_\infty, \tag{80}$$
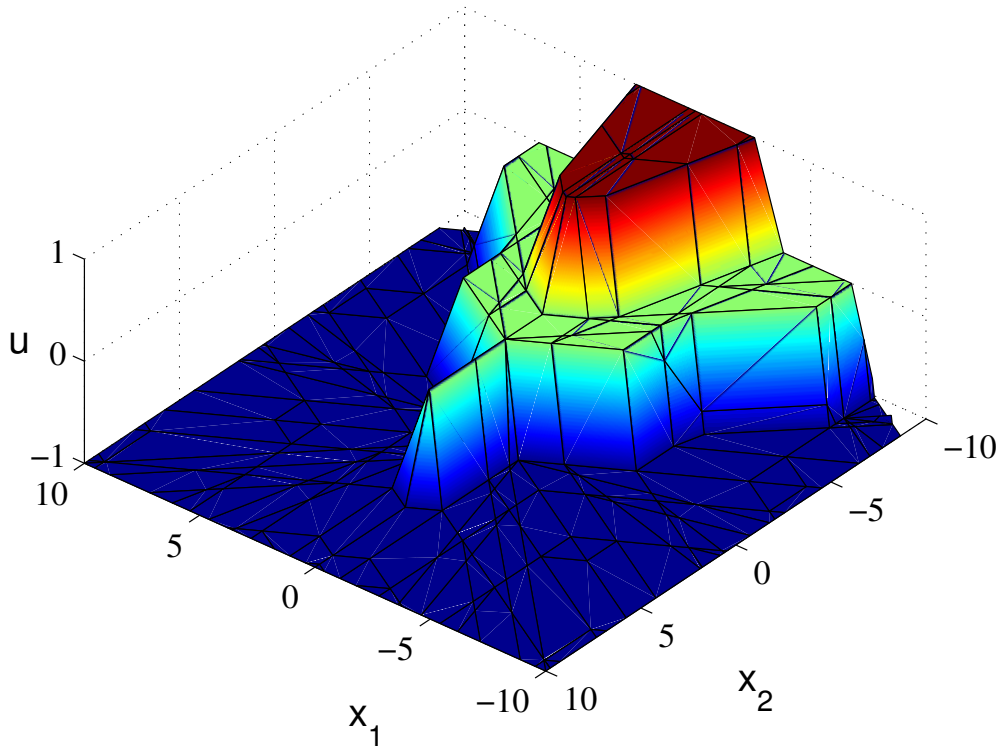
Fig. 3. The optimal control law $\boldsymbol{u}$.

because of Lipschitz continuity of $V$, where $\sigma_c$ can be computed as in (51). Adding (79) and (80) yields

$$V(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{e})) - V(\boldsymbol{x}) \leq -\gamma_c \|\boldsymbol{x}\|_\infty + \sigma_c \|\boldsymbol{e}\|_\infty, \tag{81}$$

with $\gamma_c = 1$ (since $\boldsymbol{Q} = \boldsymbol{I}$) and $\sigma_c = 4.23$. Hence, we can make use of Lemma 2 and the procedure described in Section V (in short, solving (41)). Hereto, we program Algorithm V-G in Matlab with $\rho_{\max} = 0.23 < \frac{\gamma_c}{\sigma_c}$ and make use of Yalmip [50] (version R14SP3) and glpkmex [51] (version 2.8) as interfaces to the GLPK linear solver library [44] (version 4.38). We use a *rectangular* partition and binary refinement procedure (meaning that each rectangle is split in $2^2$ equally sized rectangles at the refinement step (step 12) of the algorithm in Subsection V-G).

The resulting approximate control law $\tilde{\boldsymbol{u}}$ is displayed in Figure 4. It was calculated in 154 sec. (on a single core of an Intel Core 2 Duo P8400, running 64-bit versions of Ubuntu 10.04 and Matlab R2009a), and has 55 regions over 5 levels of refinement. To verify the performance of this approximate control, simulations were performed with a starting point $\boldsymbol{x}_0 = \begin{bmatrix} 9.97 & 9.97 \end{bmatrix}^\top$ close to the boundary of $\mathbb{X}_f$, as displayed in Figure 5. As can be seen, the closed-loop system responses of the high- and low-complexity controller are similar and converge to the origin while remaining within the constraints, as guaranteed by our theory. Note that the original optimal control law has 57 irregular regions (after simplification), whereas the approximate PWA state feedback has 55 regular regions. Hence, the number of regions is comparable, but the approximate control law has the advantage of regular regions being easier to implement on-line. To validate and demonstrate the latter statement,
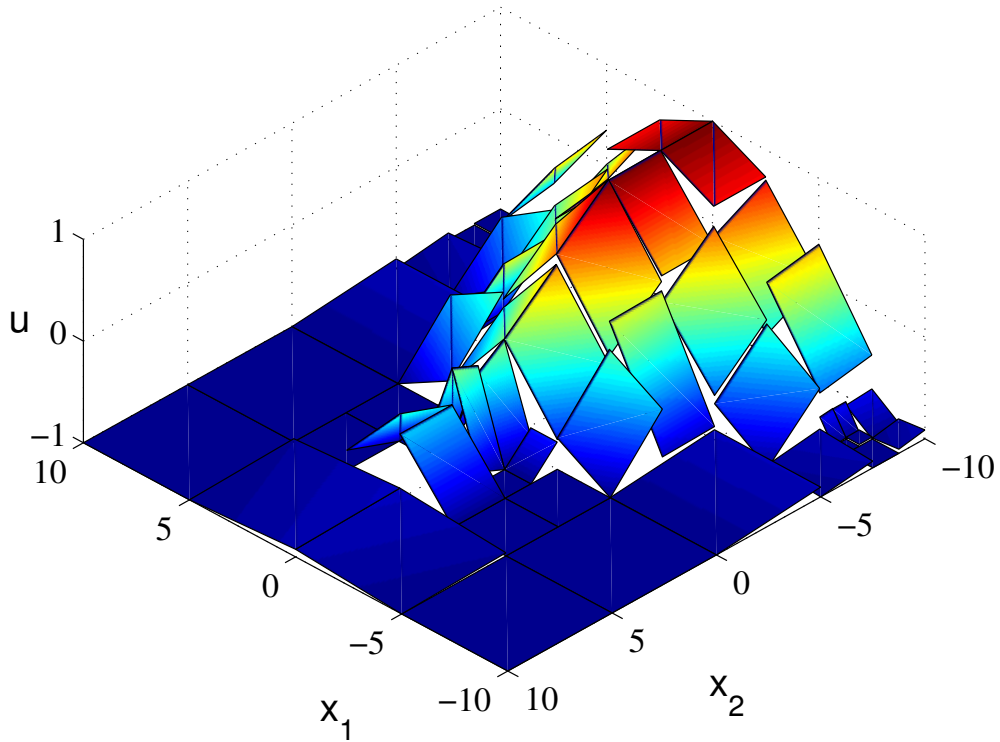
Fig. 4. The approximate control law $\tilde{u}$.

simulations are performed with Xilinx ISE 12.3 software for a Spartan 3 field-programmable gate array (FPGA). The optimal control law with 57 regions is implemented using ideas in [12], [13], resulting in a binary tree with a maximum depth of 9 levels. The approximate control law results in a quad-tree with a maximum depth of 5 levels. Moreover, the elementary operations performed at each node of the tree are two multiply-and-accumulate (MAC) and one 12-bit comparison, versus two binary comparisons, for the high– versus low–complexity control law, respectively. This results in the statistics shown in Table I, showing clearly the advantages of using the more regular partition both in real-time evaluation time as well as memory requirements. In summary, closed-loop simulations of the approximate control law show comparable behavior to those of the optimal control law (see e.g. Figure 5), while leading to a four times faster on-line evaluation needing less than half the amount of memory. It is expected that for problems of higher dimension the reduction in on-line evaluation time will be even larger, thereby indicating the relevance of our method as an enabling step to bring explicit MPC also to higher-dimensional fast applications.

*Remark 16:* While this comparison is based on FPGA implementation, similar advantages can be readily expected for implementation on application-specific integrated circuit (ASIC)–based platforms, but also for CPU– or GPU–based platforms, since the main rationale behind regular partitions in terms of efficient point location and storage also applies there.
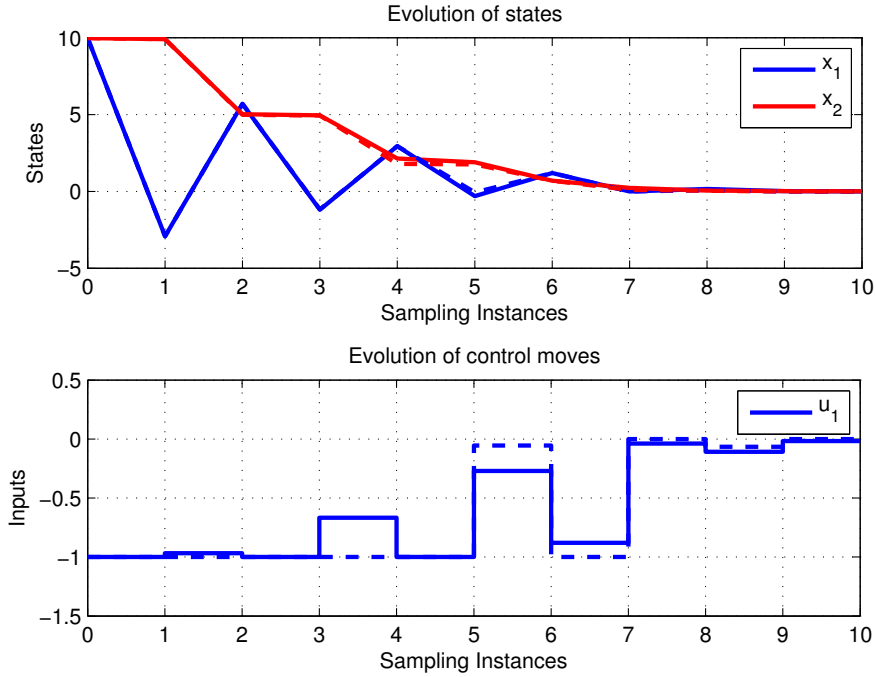
Fig. 5. Simulation of closed-loop ($\tilde{u}$ solid, $u$ dashed)

| Implementation | # clock cycles | # 12-bit memory cells |
|---|---|---|
| Exact optimal ($u$) | 36 | 396 |
| Approximate ($\tilde{u}$) | 9 | 165 |

TABLE I

COMPARISON OF SIMULATIONS OF FPGA IMPLEMENTATIONS

## VIII. CONCLUSION

In this paper we have presented a novel approximation method for PWA control laws converting high-complexity controllers into low-complexity controllers. As a particular application domain of the presented techniques we envision explicit MPC controllers that under certain conditions result in PWA control laws having general irregular partitions, which are often prohibitively complex (in terms of on-line computations and/or memory required for evaluation) for usage in the context of fast or large-scale systems. From an implementation point of view it is therefore desirable to have low-complexity controllers, but still having guarantees with respect to closed-loop stability and constraint satisfaction. In this paper, we therefore searched for approximate PWA controllers that are defined over partitions with less regions and/or regions with a more regular shape, enhancing fast and efficient on-line evaluation.

The main rationale behind the new approximation method presented in this paper is the concept of input-to-state stability with respect to approximation errors. Based on this concept, for which we have provided two computational methods to derive it from nominal stability, we conceived bounds on the approximation error

between the original high-complexity PWA control law and the approximate low-complexity state feedback preserving closed-loop stability. These stability bounds are converted into semidefinite programs (LMIs) if 2–norms are used and into linear programming (LP) problems in case of $1, \infty$–norms. Consequently, in the former case constrained LMI problems and in the latter LP feasibility problems have to be solved. Interestingly, the conditions have a local region-dependent character, which naturally allows for an automated refinement procedure to allow more flexible PWA control functions in regions where this is necessary. Hence, our method can start from a very coarse partition of the feasible set, which is automatically refined in regions where this is necessary to guarantee the conditions for stability and constraint satisfaction. From this perspective, our method synthesizes both the controller gains as well as the partition itself. This refinement inherently creates a multiscale partition with a corresponding search tree, which in many cases can be exploited to make point location even more efficient. As a consequence, once an ISS Lyapunov function with respect to the approximation error is available, the presented methods apply to PWA systems and PWA controllers, while still providing a priori guarantees on closed-loop stability and constraint satisfaction. A numerical example illustrated the main aspects of our new approximation method.

Future work involves the question how to obtain PWA controllers that are ISS with respect to control approximation errors by direct design (as opposed to exploiting inherent ISS as obtained from nominal stability). First ideas for answering these questions have already been noted in Remark 2. In addition, user-friendly numerical tools will be developed in order to implement the proposed methods in an efficient manner.

## REFERENCES

[1] A. Hassibi and S. Boyd, "Quadratic stabilization and control of piecewise-linear systems," in *Proc. 1998 American Control Conference*, vol. 6, 1998, pp. 3659–3664.

[2] A. Rantzer and M. Johansson, "Piecewise linear quadratic optimal control," *IEEE Transactions on Automatic Control*, vol. 45, no. 4, pp. 629–637, 2000.

[3] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.

[4] M. Baotić, F. J. Christophersen, and M. Morari, "Constrained optimal control of hybrid systems with a linear performance index," *IEEE Trans. Aut. Control*, vol. 51, no. 12, pp. 1903–1919, 2006.

[5] M. Lazar, W. P. M. H. Heemels, S. Weiland, and A. Bemporad, "Stabilizing model predictive control of hybrid systems," *IEEE Transactions on Automatic Control*, vol. 51, no. 11, pp. 1813–1818, 2006.

[6] E. F. Camacho and C. Bordons, *Model Predictive Control*, 2nd ed., ser. Advanced Textbooks in Control and Signal Processing. London: Springer-Verlag, 2004.

[7] J. Maciejowski, *Predictive control with constraints*. Harlow, UK: Prentice Hall, 2002.

[8] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control* : Theory and Design, 1st ed. Madison, Wisconsin, USA: Nob Hill Publishing, 2009.

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

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

[11] M. de Berg, M. van Kreveld, M. Overmars, and O. Cheong, *Computational Geometry*, 3rd ed. Berlin: Springer-Verlag, 2008.

[12] P. Tøndel, T. A. Johansen, and A. Bemporad, "Evaluation of piecewise affine control via binary search tree," *Automatica*, vol. 39, no. 5, pp. 945–950, 2003.

[13] A. Oliveri, A. Oliveri, T. Poggi, and M. Storace, "Circuit implementation of piecewise-affine functions based on a binary search tree," in *2009 ECCTD*, 2009, pp. 145–148.

[14] F. Bayat, T. A. Johansen, and A. A. Jalali, "Using hash tables to manage the time-storage complexity in a point location problem: Application to explicit model predictive control," *Automatica*, vol. 47, no. 3, pp. 571–577, 2011.

[15] A. Bemporad and C. Filippi, "Suboptimal explicit receding horizon control via approximate multiparametric quadratic programming," *Journal of Optimization Theory and Applications*, vol. 117, no. 1, pp. 9–38, 2003.

[16] T. A. Johansen and A. Grancharova, "Approximate explicit constrained linear model predictive control via orthogonal search tree," *IEEE Trans. Aut. Control*, vol. 48, no. 5, pp. 810–815, may 2003.

[17] D. Muñoz de la Peña, A. Bemporad, and C. Filippi, "Robust explicit MPC based on approximate multiparametric convex programming," *IEEE Trans. Aut. Control*, vol. 51, no. 8, pp. 1399–1403, 2006.

[18] F. J. Christophersen, M. N. Zeilinger, C. N. Jones, and M. Morari, "Controller complexity reduction for piecewise affine systems through safe region elimination," in *2007 46th IEEE CDC*, 2007, pp. 4773–4778.

[19] M. Kvasnica, F. J. Christophersen, M. Herceg, and M. Fikar, "Polynomial approximation of closed-form MPC for piecewise affine systems," in *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, 2008, pp. 3877–3882.

[20] S. Summers, C. N. Jones, J. Lygeros, and M. Morari, "A multiscale approximation scheme for explicit model predictive control with stability, feasibility, and performance guarantees," in *Proc. 48th IEEE CDC*, dec 2009, pp. 6327–6332.

[21] C. N. Jones and M. Morari, "Polytopic approximation of explicit model predictive controllers," *IEEE Trans. Aut. Control*, vol. 55, no. 11, pp. 2542–2553, 2010.

[22] A. Bemporad, A. Oliveri, T. Poggi, and M. Storace, "Ultra-fast stabilizing model predictive control via canonical piecewise affine approximations," *IEEE Trans. Aut. Control*, 2010, in press.

[23] C. N. Jones and M. Morari, "Approximate Explicit MPC using bilevel optimization," in *European Control Conference*, Budapest, Hungary, aug 2009.

[24] P. Julián, A. Desages, and O. Agamennoni, "High-level canonical piecewise linear representation using a simplicial partition," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 46, no. 4, pp. 463–480, 1999.

[25] M. Storace, L. Repetto, and M. Parodi, "A method for the approximate synthesis of cellular non-linear networks – part 1: Circuit definition," *International Journal of Circuit Theory and Applications*, vol. 31, no. 3, pp. 277–297, 2003.

[26] M. Storace and T. Poggi, "Digital architectures realizing piecewise-linear multivariate functions: Two FPGA implementations," *Int. J. Circuit Theory Appl.*, vol. 39, no. 1, pp. 1–15, 2009.

[27] M. di Federico, T. Poggi, P. Julián, and M. Storace, "Integrated circuit implementation of multi-dimensional piecewise-linear functions," *Digital Signal Processing*, vol. 20, no. 6, pp. 1723–1732, 2010.

[28] E. D. Sontag, "Smooth stabilization implies coprime factorization," *IEEE Trans. Aut. Control*, vol. 34, no. 4, pp. 435–443, 1989.

[29] ——, "Further facts about ISS," *IEEE Trans. Aut. Control*, vol. 35, no. 4, pp. 473–476, 1990.

[30] Z. P. Jiang and Y. Wang, "Input-to-state stability for discrete-time nonlinear systems," *Automatica*, vol. 37, no. 6, pp. 857–869, 2001.

[31] M. Lazar, D. Muñoz de la Peña, W. Heemels, and T. Alamo, "On input-to-state stability of min–max nonlinear model predictive control," *Systems & Control Letters*, vol. 57, no. 1, pp. 39–48, 2008.

[32] B. J. P. Roset, W. P. M. H. Heemels, M. Lazar, and H. Nijmeijer, "On robustness of constrained discrete-time systems to state measurement errors," *Automatica*, vol. 44, no. 4, pp. 1161–1165, 2007.

[33] M. J. Messina, S. E. Tuna, and A. R. Teel, "Discrete-time certainty equivalence output feedback: allowing discontinuous control laws including those from model predictive control," *Automatica*, vol. 41, no. 4, pp. 617–628, 2005.

[34] D. L. Marruedo, T. Alamo, and E. F. Camacho, "Input-to-state stable MPC for constrained discrete-time nonlinear systems with bounded additive uncertainties," in *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, vol. 4, 2002, pp. 4619–4624.

[35] G. Grimm, M. J. Messina, S. E. Tuna, and A. R. Teel, "Nominally robust model predictive control with state constraints," *IEEE Transactions on Automatic Control*, vol. 52, no. 10, pp. 1856–1870, 2007.

[36] M. Lazar and W. P. M. H. Heemels, "Predictive control of hybrid systems: Input-to-state stability results for sub-optimal solutions," *Automatica*, vol. 45, no. 1, pp. 180–185, 2009.

[37] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in system and control theory*, 1st ed., ser. Studies in Applied Mathematics.   Philadelphia: Society for Industrial and Applied Mathematics, 1994, vol. 15.

[38] M. Johansson and A. Rantzer, "Computation of piecewise quadratic Lyapunov functions for hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 555–559, 1998.

[39] 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, 2002.

[40] M. Hovd and S. Olaru, "Piecewise quadratic Lyapunov functions for stability verification of approximate explicit MPC," *Modeling, Identification and Control: A Norwegian Research Bulletin*, vol. 31, no. 2, pp. 45–53, 2010.

[41] S. Boyd and L. Vandenberghe, *Convex Optimization*, 7th ed. Cambridge University Press, 2009.

[42] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11–12, pp. 625–653, 1999.

[43] I. Corp., "ILOG CPLEX V12.1 User's Manual," 2009. [Online]. Available: www.ibm.com/cplex

[44] A. Makhorin, "GLPK (GNU Linear Programming Kit) User's Guide," 2009. [Online]. Available: http://www.gnu.org/software/glpk/

[45] Y. Ohta and H. Yokoyama, "Stability analysis of uncertain piecewise linear systems using piecewise quadratic Lyapunov functions," in *2010 IEEE ISIC*, 2010, pp. 2112–2117.

[46] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[47] M. Lazar, W. P. M. H. Heemels, and A. R. Teel, "Lyapunov functions, stability and input-to-state stability subtleties for discrete-time discontinuous systems," *IEEE Trans. Aut. Control*, vol. 54, no. 10, pp. 2421–2425, 2009.

[48] F. Borrelli, M. Baotić, A. Bemporad, and M. Morari, "Dynamic programming for constrained optimal control of discrete-time linear hybrid systems," *Automatica*, vol. 41, no. 10, pp. 1709–1721, 2005.

[49] M. Kvasnica, P. Grieder, and M. Baotić, "Multi-Parametric Toolbox (MPT)," March 2006. [Online]. Available: http://control.ee.ethz.ch/ mpt/

[50] J. Löfberg, "YALMIP: A Toolbox for Modeling and Optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[51] N. Giorgetti and N. Klitgord, "GLPKMEX, a Matlab MEX interface for the GLPK library," 2009. [Online]. Available: http://glpkmex.sourceforge.net