Available online at www.sciencedirect.com
SCIENCE DIRECT®

automatica

Automatica 41 (2005) 1709–1721

www.elsevier.com/locate/automatica

Dynamic programming for constrained optimal control of discrete-time linear hybrid systems ☆

Francesco Borrelli[a,*], Mato Baotić[a,1], Alberto Bemporad[b], Manfred Morari[a]

[a]*Automatic Control Laboratory, ETH Zentrum, CH-8092 Zurich, Switzerland*
[b]*Dip. Ingegneria dell'Informazione, University of Siena, 53100 Siena, Italy*

Received 8 July 2003; received in revised form 11 February 2005; accepted 19 April 2005
Available online 21 June 2005

Abstract

In this paper we study the solution to optimal control problems for constrained discrete-time linear hybrid systems based on quadratic or linear performance criteria. The aim of the paper is twofold. First, we give basic theoretical results on the structure of the optimal state-feedback solution and of the value function. Second, we describe how the state-feedback optimal control law can be constructed by combining multiparametric programming and dynamic programming.

Keywords: Hybrid systems; Piecewise affine systems; Optimal control; Dynamic programming; Multiparametric programming

1. Introduction

Recent technological innovations have caused a considerable interest in the study of dynamical processes of a mixed continuous and discrete nature, denoted as hybrid systems. In their most general form hybrid systems are characterized by the interaction of continuous-time models (governed by differential or difference equations), and of logic rules and discrete event systems (described, for example, by temporal logic, finite state machines, if-then-else rules) and discrete components (on/off switches or valves, gears or speed selectors, etc.). Such systems can switch between many operating modes where each mode is governed by its own characteristic dynamical laws. Mode transitions are triggered by variables crossing specific thresholds (state events), by the lapse of certain time periods (time events), or by external inputs (input events) (Antsaklis, 2000). A detailed discussion of different modeling frameworks for hybrid systems that appeared in the literature goes beyond the scope of this paper; the main concepts can be found in Antsaklis (2000), Branicky, Borkar, and Mitter (1998), Bemporad and Morari (1999), Lygeros, Tomlin, and Sastry (1999).

Different methods for the analysis and design of controllers for hybrid systems have emerged over the last few years (Sontag, 1981; Lygeros et al., 1999; Bemporad & Morari, 1999). Among them, the class of optimal controllers is one of the most studied. The approaches differ greatly in the hybrid models adopted, in the formulation of the optimal control problem and in the method used to solve it.

In this paper we focus on discrete-time linear hybrid models. In our hybrid modeling framework we allow (i) the system to be discontinuous, (ii) both states and inputs to assume continuous and discrete values, (iii) events to be both internal, i.e., caused by the state reaching a particular boundary, and exogenous, i.e., forced by a switch to some other operating mode, and (iv) states and inputs to fulfill

☆ This paper was not presented at any IFAC meeting. This paper was recommended for publication in revised form by Associate Editor Andrew R. Teel under the direction of Editor H.K. Khalil.

* Corresponding author: Dipartimento di Ingegneria, Università del Sannio in Benevento, Piazza Roma 21, 82100 Benevento, Italy. Tel.: +39 0824 30 5586; fax: +39 0824 32 5246.

E-mail addresses: francesco.borrelli@unisannio.it (F. Borrelli), mato.baotic@fer.hr (M. Baotić), bemporad@dii.unisi.it (A. Bemporad), morari@control.ee.ethz.ch (M. Morari).

[1] Current address: University of Zagreb, 10000 Zagreb, Croatia.

0005-1098/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2005.04.017

linear constraints. We will focus on discrete-time piecewise affine (PWA) models. Discrete-time PWA models can describe a large number of processes, such as discrete-time linear systems with static piecewise-linearities; discrete-time linear systems with discrete states and inputs; switching systems where the dynamic behavior is described by a finite number of discrete-time linear models together with a set of logic rules for switching among these models; approximation of nonlinear discrete-time dynamics, e.g., via multiple linearizations at different operating points.

In discrete-time hybrid systems an event can occur only at instants that are multiples of the sampling time, and many interesting mathematical phenomena occurring in continuous-time hybrid systems such as Zeno behaviors do not exist. However, the solution to optimal control problems is still complex: the solution to the HJB equation can be discontinuous and the number of possible switches grows exponentially with the length of the horizon of the optimal control problem. Nevertheless, we will show that for the class of linear discrete-time hybrid systems we can *characterize* and *compute* the optimal control law exactly *without gridding* the state space.

The solution to optimal control problems for discrete-time hybrid systems was first outlined by Sontag (1981). In his plenary presentation (Mayne, 2001) at the 2001 European Control Conference, Mayne presented an intuitively appealing characterization of the state-feedback solution to optimal control problems for linear hybrid systems with performance criteria based on quadratic and linear norms. The detailed exposition presented in the initial part of this paper follows a similar line of argumentation and shows that the state-feedback solution to the finite time optimal control problem is a time-varying PWA feedback control law, possibly defined over non-convex regions. Moreover, we give insight into the structure of the optimal state-feedback solution and of the value function.

In the second part of the paper we describe how the optimal control law can be efficiently computed by means of multiparametric programming. In particular, we propose a novel algorithm that solves the Hamilton–Jacobi–Bellman equation by using a simple multiparametric solver. In collaboration with different companies and institutes, the results described in this paper have been applied to a wide range of problems (Baotic, Vasak, Morari, & Peric, 2003; Bemporad, Borodani, & Mannelli, 2003; Bemporad, Giorgetti, Kolmanovsky, & Hrovat, 2002; Bemporad & Morari, 1999; Borrelli, Bemporad, Fodor, & Hrovat, 2001; Ferrari-Trecate et al., 2002; Mignone, 2002; Möbus, Baotic, & Morari, 2003; Torrisi & Bemporad, 2004). Simple examples that highlight the main features of the hybrid system approach presented in this paper can be found in Borrelli, Baotic, Bemporad, and Morari (2003).

Before formulating optimal control problems for hybrid systems we will give a short overview on multiparametric programming and on discrete-time linear hybrid systems.

2. Definitions and basic results

We will use the following non-standard definitions:

Definition 1. A polyhedron is a set that equals the intersection of a finite number of closed halfspaces. An open set $\mathscr{R}$ whose closure $\bar{\mathscr{R}}$ is a polyhedron is called open polyhedron. A "neither open nor closed polyhedron" is a neither open nor closed set $\mathscr{R}$ whose closure $\bar{\mathscr{R}}$ is a polyhedron. A non-Euclidean polyhedron is a set whose closure equals the union of a finite number of polyhedra.

Definition 2. A collection of sets $\mathscr{R}_1, \ldots, \mathscr{R}_N$ is a *partition* of a set Θ if (i) $\bigcup_{i=1}^N \mathscr{R}_i = \Theta$, (ii) $\mathscr{R}_i \cap \mathscr{R}_j = \emptyset$, $\forall i \neq j$. Moreover $\mathscr{R}_1, \ldots, \mathscr{R}_N$ is a *polyhedral partition* of a polyhedral set Θ if $\mathscr{R}_1, \ldots, \mathscr{R}_N$ is a partition of Θ and the $\bar{\mathscr{R}}_i$'s are polyhedral sets, where $\bar{\mathscr{R}}_i$ denotes the closure of the set $\mathscr{R}_i$.

Definition 3. A function $h : \Theta \to \mathbb{R}^k$, where $\Theta \subseteq \mathbb{R}^s$, is *PWA* if there exists a partition $\mathscr{R}_1, \ldots, \mathscr{R}_N$ of Θ and $h(\theta) = H^i\theta + k^i$, $\forall \theta \in \mathscr{R}_i$, $i = 1, \ldots, N$.

Definition 4. A function $h : \Theta \to \mathbb{R}^k$, where $\Theta \subseteq \mathbb{R}^s$, is *PWA on polyhedra* (*PPWA*) if there exists a polyhedral partition $\mathscr{R}_1, \ldots, \mathscr{R}_N$ of Θ and $h(\theta) = H^i\theta + k^i$, $\forall \theta \in \mathscr{R}_i$, $i = 1, \ldots, N$.

Piecewise quadratic (PWQ) functions and piecewise quadratic functions on polyhedra (PPWQ) are defined analogously.

Definition 5. A function $q : \Theta \to \mathbb{R}$, where $\Theta \subseteq \mathbb{R}^s$, is a *multiple quadratic function* of multiplicity $d \in \mathbb{N}^+$ if $q(\theta) = \min\{q^1(\theta) \triangleq \theta' Q^1 \theta + l^1\theta + c^1, \ldots, q^d(\theta) \triangleq \theta' Q^d \theta + l^d\theta + c^d\}$, $Q^i > 0$, $\forall i = 1, \ldots, d$ and Θ is a convex polyhedron.

Definition 6. A function $q : \Theta \to \mathbb{R}$, where $\Theta \subseteq \mathbb{R}^s$, is a *multiple PWQ on polyhedra* (multiple PPWQ) if there exists a polyhedral partition $\mathscr{R}_1, \ldots, \mathscr{R}_N$ of Θ and $q(\theta) = \min\{q_i^1(\theta) \triangleq \theta' Q_i^1 \theta + l_i^1\theta + c_i^1, \ldots, q_i^{d_i}(\theta) \triangleq \theta' Q_i^{d_i} \theta + l_i^{d_i}\theta + c_i^{d_i}\}$, $\forall \theta \in \mathscr{R}_i$, $i = 1, \ldots, N$. We define d_i to be the multiplicity of the function q in the polyhedron $\mathscr{R}_i$, and $d = \sum_{i=1}^N d_i$ to be the multiplicity of the function q. (Note that Θ is not necessarily convex.)

3. Basics of multiparametric programming

Consider the nonlinear mathematical program dependent on a parameter vector x appearing in the cost function and in the constraints

$$
\begin{aligned}
J^*(x) = \inf_z \ & f(z, x) \\
\text{subj. to } & g(z, x) \leqslant 0 \\
& z \in M,
\end{aligned} \tag{1}
$$

where $z \in \mathbb{R}^s$ is the optimization vector, $x \in \mathbb{R}^n$ is the parameter vector, $f : \mathbb{R}^s \times \mathbb{R}^n \to \mathbb{R}$ is the cost function, $g : \mathbb{R}^s \times \mathbb{R}^n \to \mathbb{R}^{n_g}$ are the constraints and $M \subseteq \mathbb{R}^s$.

A small perturbation of the parameter x in (1) can cause a variety of outcomes, i.e., depending on the properties of the functions f and g the solution $z^*(x)$ may vary smoothly or change abruptly as a function of x. We denote by K^* the set of feasible parameters, i.e.,

$$K^* = \{x \in \mathbb{R}^n \mid \exists z \in M, g(z, x) \leqslant 0\}, \tag{2}$$

by $R : \mathbb{R}^n \to 2^{\mathbb{R}^s}$, where $2^{\mathbb{R}^s}$ denotes the set of all subsets of $\mathbb{R}^s$, the point-to-set map that assigns the set of feasible z

$$R(x) = \{z \in M \mid g(z, x) \leqslant 0\} \tag{3}$$

to a parameter x, by $J^* : K^* \to \mathbb{R} \cup \{-\infty\}$ the real-valued function which expresses the dependence on x of the minimum value of the objective function over K^*, i.e.,

$$J^*(x) = \inf_z \{f(z, x) \mid x \in K^*,\ z \in R(x)\}, \tag{4}$$

and by $Z^* : K^* \to 2^{\mathbb{R}^s}$ the point-to-set map which expresses the dependence on x of the set of optimizers, i.e, $Z^*(\bar{x})=\{z \in R(\bar{x}) | f(z, \bar{x}) = J^*(\bar{x})\}$ with $\bar{x} \in K^*$.

$J^*(x)$ will be referred to as the optimal value function or simply *value function*, $Z^*(x)$ will be referred to as the *optimal set*. We will denote by $z^* : \mathbb{R}^n \to \mathbb{R}^s$ one of the possible single valued functions that can be extracted from Z^*, and z^* will be called the *optimizer function*. If $Z^*(x)$ is a singleton for all x, then $z^*(x)$ is the only element of $Z^*(x)$.

Our interest in problem (1) will become clear in the following sections. We can anticipate here that optimal control problems for nonlinear systems can be reformulated as the mathematical program (1) where z is the input sequence to be optimized and x the initial state of the system. Therefore, the study of the properties of J^* and Z^* is fundamental for the study of properties of state-feedback optimal controllers.

Fiacco (1983, Chapter 2) provides conditions under which the solution of nonlinear multiparametric programs (1) is locally well behaved and establishes properties of the solution as a function of the parameters. In the following we report a basic result (Hogan, 1973) which focuses on a restricted set of functions $f(z, x)$ and $g(z, x)$:

Theorem 1 (*Hogan, 1973*). *Consider the multiparametric nonlinear program* (1). *Assume that M is a convex and bounded set in $\mathbb{R}^s$, f is continuous and the components of g are convex on $M \times \mathbb{R}^n$. Then, $J^*(x)$ is continuous at each $x \in K^*$.*

Unfortunately very little can be said without continuity assumption on f and convexity assumption on g. Below we restrict our attention to two special classes of multiparametric programming.

3.1. Multiparametric quadratic program

Consider the multiparametric program

$$J^*(x) = \tfrac{1}{2}x'Yx + \min_z \ \tfrac{1}{2}z'Hz + z'Fx \quad \text{subj. to} \quad Cz \leqslant c + Sx, \tag{5}$$

where $z \in \mathbb{R}^{n_z}$ is the optimization vector, $x \in \mathbb{R}^n$ is the vector of parameters, and $C \in \mathbb{R}^{q \times n_z}$, $c \in \mathbb{R}^q$, $S \in \mathbb{R}^{q \times n}$ are constant matrices. We refer to the problem of computing $z^*(x)$ and $J^*(x)$ in (5) as (right-hand side) *multiparametric quadratic program* (mp-QP).

Theorem 2 (*Bemporad, Morari, Dua, & Pistikopoulos, 2002*). *Consider the mp-QP* (5). *Assume $H \succ 0$ and $\begin{bmatrix} Y & F' \\ F & H \end{bmatrix} \succcurlyeq 0$. The set K^* is a polyhedral set, the value function $J^* : K^* \to \mathbb{R}$ is PPWQ, convex and continuous and the optimizer $z^* : K^* \to \mathbb{R}^{n_z}$ is PPWA and continuous.*

4. Hybrid systems

Several modeling frameworks have been introduced for discrete-time hybrid systems. Among them, *PWA* systems (Sontag, 1981) are defined by partitioning the state space into polyhedral regions and associating with each region a different affine state-update equation

$$x(t+1) = A^i x(t) + B^i u(t) + f^i \quad \text{if} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \in \mathscr{P}^i,\ i = \{1, \dots, s\}, \tag{6}$$

where $x \in \mathbb{R}^{n_c} \times \{0,1\}^{n_\ell}$, $u \in \mathbb{R}^{m_c} \times \{0,1\}^{m_\ell}$, $\{\mathscr{P}^i\}_{i=1}^s$ is a polyhedral partition of the set of the state+input space $\mathscr{P} \subset \mathbb{R}^{n+m}$, $n \triangleq n_c + n_\ell$, $m \triangleq m_c + m_\ell$. We denote by $x_c \in \mathbb{R}^{n_c}$ and $u_c \in \mathbb{R}^{m_c}$ the real components of the state and input vector, respectively. We will give the following definitions of continuous PWA system.

Definition 7. We say that the PWA system (6) is *continuous* if the mapping $(x_c(t), u_c(t)) \mapsto x_c(t+1)$ is continuous and $n_\ell = m_\ell = 0$. The PWA system (6) is *continuous in the real input space* if the mapping $(x_c(t), u_c(t)) \mapsto x_c(t+1)$ is continuous w.r.t. u_c. Analogously, we define PWA systems continuous in the real state space.

Our main motivation for focusing on discrete-time models stems from the need to analyze these systems and to solve optimization problems, such as optimal control or scheduling problems, for which the continuous time counterpart would not be easily computable.

PWA systems are equivalent to interconnections of linear systems and finite automata. In Heemels, De Schutter, and Bemporad (2001) the authors have proven the equivalence of linear discrete-time PWA systems and other classes of discrete-time hybrid systems. PWA models can be generated automatically through appropriate conversion procedures (Bemporad, 2004) from discrete hybrid automata, a very general class of linear hybrid systems that can be modeled in the language HYSDEL (Torrisi & Bemporad, 2004).

5. Problem formulation

Consider the PWA system (6) subject to hard input and state constraints

$$Ex(t) + Lu(t) \leqslant M_c \tag{7}$$

for $t \geqslant 0$, and denote by *constrained PWA system* (CPWA) the restriction of the PWA system (6) over the set of states and inputs defined by (7),

$$x(t+1) = A^i x(t) + B^i u(t) + f^i \quad \text{if } \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \in \tilde{\mathscr{P}}^i, \tag{8}$$

where $\{\tilde{\mathscr{P}}^i\}_{i=1}^s$ is the new polyhedral partition of the sets of state+input space $\mathbb{R}^{n+m}$ obtained by intersecting the sets $\mathscr{P}^i$ in (6) with the polyhedron described by (7). The union of the polyhedral partitions $\tilde{\mathscr{P}} \triangleq \bigcup_{i=1}^s \tilde{\mathscr{P}}^i$ will implicitly define the feasible region R_{feas} in the input space as a function of x:

$$R_{\text{feas}}(x) = \{u \in \mathbb{R}^{m_c} \times \{0,1\}^{m_\ell} | (x,u) \in \tilde{\mathscr{P}}\}.$$

We assume that $R_{\text{feas}}(x)$ is a compact set for any x and the following:

Assumption 1. System (8) is continuous in the real input and real state space.

Assumption 1 requires that the PWA function that defines the update of the continuous states is continuous on the boundaries of contiguous polyhedral cells, and therefore allows one to work with the closure of sets $\tilde{\mathscr{P}}^i$ without the need of introducing multi-valued state update equations. With abuse of notation in the following sections $\tilde{\mathscr{P}}^i$ will always denote the closure of $\tilde{\mathscr{P}}^i$. Discontinuous PWA systems will be discussed in Section 8.

We define the following cost function:

$$J(U_N, x(0)) \triangleq \|Px_N\|_p + \sum_{k=0}^{N-1} \|Qx_k\|_p + \|Ru_k\|_p, \tag{9}$$

and consider the constrained finite-time optimal control (CFTOC) problem

$$J_0^*(x(0)) \triangleq \min_{U_N} J(U_N, x(0)) \tag{10}$$

$$\text{subj. to} \begin{cases} x_{k+1} = A^i x_k + B^i u_k + f^i \\ \quad \text{if } \begin{bmatrix} x_k \\ u_k \end{bmatrix} \in \tilde{\mathscr{P}}^i, \; i = 1, \ldots, s \\ x_N \in \mathscr{X}_f \\ x_0 = x(0), \end{cases} \tag{11}$$

where the column vector $U_N \triangleq [u_0', \ldots, u_{N-1}']' \in \mathbb{R}^{m_c N} \times \{0,1\}^{m_l N}$ is the optimization vector, N is the optimal control horizon and $\mathscr{X}_f$ is a polyhedral terminal region. In (9), $\|Qx\|_p$ denotes the p-norm of the vector Qx if $p = 1, \infty$ or $x'Qx$ if $p = 2$. In (11) we have omitted the constraints $u_k \in R_{\text{feas}}(x_k)$, $k = 1, \ldots, N$, assuming that they are implicit in the first constraints, i.e., if there exists no $\tilde{\mathscr{P}}^i$ that contains $[\begin{smallmatrix} x_k \\ u_k \end{smallmatrix}]$ then this is an infeasible point. We will use this implicit notation throughout the paper.

Note that we distinguish between the input $u(t)$ and the state $x(t)$ of plant (8) at time t and the variables u_k and x_k of the optimization problem (11).

In the following, we will assume that $Q = Q' \succcurlyeq 0$, $R = R' \succ 0$, $P \succcurlyeq 0$, for $p = 2$, and that Q, R, P are full column rank matrices for $p = 1, \infty$. We will also denote by $\mathscr{X}_k \subseteq \mathbb{R}^{n_c} \times \{0,1\}^{n_\ell}$ the set of states x_k that are feasible for (9)–(11):

$$\mathscr{X}_k = \left\{ x \in \mathbb{R}^{n_c} \times \{0,1\}^{n_\ell} \left| \begin{array}{l} \exists\, u \in \mathbb{R}^{m_c} \times \{0,1\}^{m_\ell}, \\ \exists\, i \in \{1, \ldots, s\} \\ \begin{bmatrix} x \\ u \end{bmatrix} \in \tilde{\mathscr{P}}^i \text{ and} \\ A^i x + B^i u + f^i \in \mathscr{X}_{k+1} \end{array} \right. \right\},$$
$$k = 0, \ldots, N-1,$$
$$\mathscr{X}_N = \mathscr{X}_f. \tag{12}$$

Note that the optimizer function U_N^* may not be uniquely defined if the optimal set of problem (9)–(11) is not a singleton for some $x(0)$.

In the following we need to distinguish between optimal control based on the 2-norm and optimal control based on the 1-norm or ∞-norm.

As a last remark, we want to point out that it is almost immediate to extend the results of the following sections to different formulations of hybrid optimal control problems, such as reference tracking problems or problems where penalties for switching between two different regions of operation are weighted in the cost function.

6. Solution properties

Theorem 3. *Consider the optimal control problem* (9)–(11) *with $p = 2$ and let Assumption* 1 *hold. Then, there exists a solution in the form of a PWA state-feedback control law*

$$u_k^*(x(k)) = F_k^i x(k) + G_k^i \quad \text{if } x(k) \in \mathscr{R}_k^i, \tag{13}$$

where $\mathscr{R}_k^i$, $i = 1, \ldots, N_k$ is a partition of the set $\mathscr{X}_k$ of feasible states $x(k)$, and the closure $\bar{\mathscr{R}}_k^i$ of the sets $\mathscr{R}_k^i$ has the following form:

$$\bar{\mathscr{R}}_k^i \triangleq \{x : \; x(k)' L_k^i(j) x(k) + M_k^i(j) x(k) \leqslant N_k^i(j),$$
$$j = 1, \ldots, n_k^i\}, \quad k = 0, \ldots, N-1, \tag{14}$$

and

$$x(k+1) = A^i x(k) + B^i u_k^*(x(k)) + f^i \quad \text{if} \begin{bmatrix} x(k) \\ u_k^*(x(k)) \end{bmatrix} \in \tilde{\mathscr{P}}^i,\ i = \{1, \ldots, s\}. \tag{15}$$

Proof. The piecewise linearity of the solution was first mentioned by Sontag (1981). Mayne (2001) sketched a proof. In the following we will give the proof for $u_0^*(x(0))$; the same arguments can be repeated for $u_1^*(x(1)), \ldots, u_{N-1}^*(x(N-1))$.

Case 1: *no binary inputs and states* ($m_l = n_l = 0$).

Depending on the initial state $x(0)$ and on the input sequence $U = [u_0', \ldots, u_k']$, the state x_k is either infeasible or it belongs to a certain polyhedron $\tilde{\mathscr{P}}^i$, $k = 0, \ldots, N-1$. The number of all possible locations of the state sequence $x_0, \ldots, x_{N-1}$ is equal to s^N. Denote by $\{v_i\}_{i=1}^{s^N}$ the set of all possible switching sequences over the horizon *N*, and by v_i^k the *k*th element of the sequence v_i, i.e., $v_i^k = j$ if $x_k \in \tilde{\mathscr{P}}^j$.

Fix a certain v_i and constrain the state to switch according to the sequence v_i. problem (9)–(11) becomes

$$J_{v_i}^*(x(0)) \triangleq \min_{\{U_N\}} J(U_N, x(0)) \tag{16}$$

$$\text{subj. to} \begin{cases} x_{k+1} = A^{v_i^k} x_k + B^{v_i^k} u_k + f^{v_i^k}, \\ \begin{bmatrix} x_k \\ u_k \end{bmatrix} \in \tilde{\mathscr{P}}^{v_i^k} \\ \quad k = 0, \ldots, N-1, \\ x_N \in \mathscr{X}_f, \\ x_0 = x(0). \end{cases} \tag{17}$$

Problem (16)–(17) is equivalent to a finite-time optimal control problem for a linear time-varying system with time-varying constraints and can be solved by using the approach of Bemporad et al. (2002). The first move u_0 of its solution is the PPWA feedback control law

$$u_0^i(x(0)) = \tilde{F}^{i,j} x(0) + \tilde{G}^{i,j}, \quad \forall x(0) \in \mathscr{T}^{i,j}, \quad j = 1, \ldots, N^{ri}, \tag{18}$$

where $\mathscr{D}^i = \bigcup_{j=1}^{N^{ri}} \mathscr{T}^{i,j}$ is a polyhedral partition of the convex set $\mathscr{D}^i$ of feasible states $x(0)$ for problem (16)–(17). N^{ri} is the number of regions of the polyhedral partition of the solution and it is a function of the number of constraints in problem (16)–(17). The upper index *i* in (18) denotes that the input $u_0^i(x(0))$ is optimal when the switching sequence v_i is fixed.

The set $\mathscr{X}_0$ of all feasible states at time 0 is $\mathscr{X}_0 = \bigcup_{i=1}^{s^N} \mathscr{D}^i$ and in general it is not convex. Indeed, as some initial states can be feasible for different switching sequences, the sets $\mathscr{D}^i$, $i = 1, \ldots, s^N$, in general, can overlap. The solution $u_0^*(x(0))$ to the original problem (9)–(11) can be computed in the following way. For every polyhedron $\mathscr{T}^{i,j}$ in (18):

(1) If $\mathscr{T}^{i,j} \cap \mathscr{T}^{l,m} = \emptyset$ for all $l \neq i$, $l = 1, \ldots, s^N$, and for all $m = 1, \ldots, N^{rl}$, then the switching sequence v_i is the only feasible one for all the states belonging to $\mathscr{T}^{i,j}$ and therefore the optimal solution is given by (18), i.e.

$$u_0^*(x(0)) = \tilde{F}^{i,j} x(0) + \tilde{G}^{i,j}, \quad \forall x \in \mathscr{T}^{i,j}. \tag{19}$$

(2) If $\mathscr{T}^{i,j}$ intersects one or more polyhedra $\mathscr{T}^{l_1,m_1}$, $\mathscr{T}^{l_2,m_2}, \ldots$, the states belonging to the intersection are feasible for more than one switching sequence $v_i, v_{l_1}, v_{l_2}, \ldots$, and therefore the corresponding value functions $J_{v_i}^*, J_{v_{l_1}}^*, J_{v_{l_2}}^*, \ldots$ in (16) have to be compared in order to compute the optimal control law.
Consider the simple case when only two polyhedra overlap, i.e. $\mathscr{T}^{i,j} \cap \mathscr{T}^{l,m} \triangleq \mathscr{T}^{(i,j),(l,m)} \neq \emptyset$. We will refer to $\mathscr{T}^{(i,j),(l,m)}$ as a *double feasibility polyhedron*. For all states belonging to $\mathscr{T}^{(i,j),(l,m)}$ the optimal solution is

$$u_0^*(x(0)) = \begin{cases} \tilde{F}^{i,j} x(0) + \tilde{G}^{i,j}, & \forall x(0) \in \mathscr{T}^{(i,j),(l,m)} : \\ & J_{v_i}^*(x(0)) < J_{v_l}^*(x(0)) \\ \tilde{F}^{l,m} x(0) + \tilde{G}^{l,m}, & \forall x(0) \in \mathscr{T}^{(i,j),(l,m)} : \\ & J_{v_i}^*(x(0)) > J_{v_l}^*(x(0)) \\ \begin{cases} \tilde{F}^{i,j} x(0) + \tilde{G}^{i,j} \text{ or} \\ \tilde{F}^{l,m} x(0) + \tilde{G}^{l,m} \end{cases} & \forall x(0) \in \mathscr{T}^{(i,j),(l,m)} : \\ & J_{v_i}^*(x(0)) = J_{v_l}^*(x(0)). \end{cases} \tag{20}$$

Because $J_{v_i}^*$ and $J_{v_l}^*$ are quadratic functions of $x(0)$ on $\mathscr{T}^{i,j}$ and $\mathscr{T}^{l,m}$, respectively, we find the expression (14) of the control law domain. The sets $\mathscr{T}^{i,j} \backslash \mathscr{T}^{l,m}$ and $\mathscr{T}^{l,m} \backslash \mathscr{T}^{i,j}$ are two *single feasibility non-Euclidean polyhedra* which can be partitioned into a set of *single feasibility polyhedra*, and thus be described through (14) with $L_k^i = 0$.

In order to conclude the proof, the general case of *n* intersecting polyhedra has to be discussed. We follow three main steps. Step 1: generate one polyhedron of *n*th-ple feasibility and $2^n - 2$ polyhedra, generally non-Euclidean and possibly empty and disconnected, of single, double, $\ldots$, $(n-1)$th-ple feasibility. Step 2: the *i*th-ple feasibility non-Euclidean polyhedron is partitioned into several *i*th-ple feasibility polyhedra. Step 3: any *i*th-ple feasibility polyhedron with $i > 1$ is further partitioned into at most *i* subsets (14) where in each one of them a certain feasible value function is greater than all the others. The procedure is depicted in Fig. 1 when $n = 3$.

Case 2: *binary inputs*, $m_\ell \neq 0$.

The proof can be repeated in the presence of binary inputs, $m_\ell \neq 0$. In this case the switching sequences v_i are given by all combinations of region indices and *binary inputs*, i.e., $i = 1, \ldots, (s \cdot m_\ell)^N$. The continuous component of the optimal input is given by (19) or (20). Such an optimal continuous component of the input has an associated optimal sequence v_i, whose component provides the remaining binary components of the optimal input.

Case 3: *binary states*, $n_l \neq 0$.

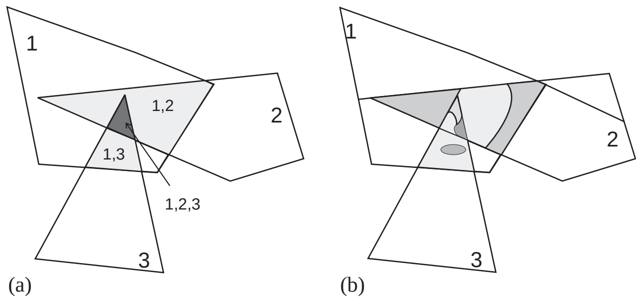

Fig. 1. Graphical illustration of the main steps for the proof of Theorem 3 when three polyhedra intersect. Step 1: the three intersecting polyhedra are partitioned into one polyhedron of triple feasibility (1,2,3), two polyhedra of double feasibility (1, 2) and (1, 3), three polyhedra of single feasibility (1), (2), (3). The sets (1), (2) and (1,2) are non-Euclidean polyhedra. Step 2: the sets (1), (2) and (1,2) are partitioned into six polyhedra of single feasibility. Step 3: value functions are compared inside the polyhedra of multiple feasibility.

The proof can be repeated in the presence of binary states by a simple enumeration of all the possible n_ℓ^N discrete state evolutions. □

From the result of the theorem above one immediately concludes that the value function J_0^* is piecewise quadratic:

$$J_0^*(x(0)) = x(0)' H_1^i x(0) + H_2^i x(0) + H_3^i \quad \text{if } x(0) \in \mathscr{R}_0^i, \tag{21}$$

The proof of Theorem 3 gives useful insights into the properties of the sets $\mathscr{R}_k^i$ in (14). We will summarize them next.

Each set $\mathscr{R}_k^i$ has an associated multiplicity j which means that j switching sequences are feasible for problem (9)–(11) starting from a state $x(k) \in \mathscr{R}_k^i$. If $j = 1$, then $\mathscr{R}_k^i$ is a polyhedron. In general, if $j > 1$ the boundaries of $\mathscr{R}_k^i$ can be described either by an affine function or by a quadratic function. In the sequel boundaries which are described by quadratic functions but degenerate to hyperplanes or sets of hyperplanes will be considered affine boundaries.

Quadratic boundaries arise from the comparison of value functions associated with feasible switching sequences, thus a maximum of $j - 1$ quadratic boundaries can be present in a j-ple feasible set. The *affine* boundaries can be of three types. Type **a**: they are inherited from the original j-ple feasible non-Euclidean polyhedron. In this case beyond such boundaries the multiplicity of the feasibility changes. Type **b**: they are artificial cuts needed to describe the original j-ple feasible non-Euclidean polyhedron as a set of j-ple feasible polyhedra. Beyond type **b** boundaries the multiplicity of the feasibility does not change. Type **c**: they arise from the comparison of quadratic value functions which degenerate in an affine boundary.

In conclusion, we can state the following proposition:

Proposition 1. *The value function* J_k^*

(1) *is a quadratic function of the states inside each* $\mathscr{R}_k^i$;
(2) *is continuous on quadratic and affine boundaries of types* **b** *and* **c**;
(3) *might be discontinuous only on affine boundaries of type* **a**;

and the optimizer u_k^*

(1) *is an affine function of the states inside each* $\mathscr{R}_k^i$;
(2) *is continuous and unique on affine boundaries of type* **b**;
(3) *is non-unique on quadratic boundaries, except possibly at isolated points*;
(4) *might be non-unique on affine boundaries of type* **c**;
(5) *might be discontinuous on affine boundaries of type* **a**.

Based on Proposition 1 one can highlight the only source of discontinuity of the value function: affine boundaries of *type a*. The following corollary gives a useful insight into the class of possible value functions.

Corollary 1. J_0^* *is a lower-semicontinuous PWQ function on* $\mathscr{X}_0$.

Proof. The proof follows from the result on the minimization of lower-semicontinuous point-to-set maps in (Berge, 1997). Below we give a simple proof without introducing the notion of point-to-set maps.

Only points where a discontinuity occurs are relevant for the proof, i.e., states belonging to boundaries of type **a**. From Assumption 1 it follows that the feasible switching sequences for a given state $x(0)$ are all the feasible switching sequences associated with any set $\mathscr{R}_0^j$ whose closure $\bar{\mathscr{R}}_0^j$ contains $x(0)$. Consider a state $x(0)$ belonging to boundaries of type **a** and the proof of Theorem 3. The only case of discontinuity can occur when (i) a j-ple feasible set $\mathscr{P}_1$ intersects an i-ple feasible set $\mathscr{P}_2$ with $i < j$, (ii) there exists a point $x(0) \in \mathscr{P}_1$, $\mathscr{P}_2$ and a neighbor $\mathscr{N}(x(0))$ with $x, y \in \mathscr{N}(x(0))$, $x \in \mathscr{P}_1$, $x \notin \mathscr{P}_2$ and $y \in \mathscr{P}_2$, $y \notin \mathscr{P}_1$. The proof follows from the previous statements and the fact that $J_0^*(x(0))$ is the minimum of all $J_{v_i}^*(x(0))$ for all feasible switching sequences v_i. □

The result of Corollary 1 will be extensively used in the following sections. Even if value function and optimizer are discontinuous, one can work with the closure $\bar{\mathscr{R}}_k^j$ of the original sets $\mathscr{R}_k^j$ without explicitly considering their boundaries. In fact, if a given state $x(0)$ belongs to several regions $\bar{\mathscr{R}}_0^1, \ldots, \bar{\mathscr{R}}_0^p$, then the minimum value among the optimal values (21) associated with each region $\bar{\mathscr{R}}_0^1, \ldots, \bar{\mathscr{R}}_0^p$ allows us to identify the region of the set $\mathscr{R}_0^1, \ldots, \mathscr{R}_0^p$ containing $x(0)$.

Next we show some interesting properties of the optimal control law when we restrict our attention to smaller classes of PWA systems.

Corollary 2. *Assume that the PWA system* (8) *is continuous, and that* $E=0$ *in* (7) *and* $\mathcal{X}_f=\mathbb{R}^n$ *in* (11) (*which means that there are no state constraints, i.e.,* $\tilde{P}$ *is unbounded in the x-space*). *Then, the value function* J_0^* *in* (11) *is continuous*.

Proof. Problem (9)–(11) becomes a multiparametric program with only input constraints when the state at time k is expressed as a function of the state at time 0 and the input sequence $u_0,\dots,u_{k-1}$, i.e., $x_k=f_{\mathrm{PWA}}((\cdots(f_{\mathrm{PWA}}(x_0,u_0),u_1),\dots,u_{k-2}),u_{k-1})$. J in (9) will be a continuous function of x_0 and $u_0,\dots,u_{N-1}$ since it is the composition of continuous functions. The input constraints on $u_0,\dots,u_{N-1}$ are convex by assumption. The proof follows from the continuity of J and Theorem 1. □

Note that $E=0$ is a sufficient condition for ensuring that constraints (7) are convex in the optimization variables $u_0,\dots,u_n$. In general, even for continuous PWA systems with state constraints it is difficult to find weak assumptions ensuring the continuity of the value function J_0^*. Ensuring the continuity of the optimal control law $u(k)=u_k^*(x(k))$ is even more difficult. A list of sufficient conditions for U_N^* to be continuous can be found in Fiacco (1976). In general, they require the convexity (or a relaxed form of it) of the cost $J(U_N,x(0))$ in U_N for each $x(0)$ and the convexity of the constraints in (11) in U_N for each $x(0)$. Such conditions are clearly very restrictive since the cost and the constraints in problem (11) are a composition of quadratic and linear functions, respectively, with the PWA dynamics of the system.

The next theorem provides a condition under which the solution $u_k^*(x(k))$ of the optimal control problem (9)–(11) is a PPWA state-feedback control law.

Theorem 4. *Assume that the optimizer* $U_N^*(x(0))$ *of* (9)–(11) *is unique for all* $x(0)$. *Then the solution to the optimal control problem* (9)–(11) *is a PPWA state-feedback control law of the form*

$$u_k^*(x(k))=F_k^i x(k)+G_k^i \quad \text{if } x(k)\in\mathcal{R}_k^i \quad k=0,\dots,N-1, \tag{22}$$

where $\mathcal{R}_k^i,\ i=1,\dots,N_k^r$, *is a polyhedral partition of the set* $\mathcal{X}_k$ *of feasible states* $x(k)$.

Proof. In Proposition 1 we concluded that the value function $J_0^*(x(0))$ is continuous on quadratic type boundaries. By hypothesis, the optimizer $u_0^*(x(0))$ is unique. Theorem 3 implies that $\tilde{F}^{i,j}x(0)+\tilde{G}^{i,j}=\tilde{F}^{l,m}x(0)+\tilde{G}^{l,m}$, $\forall x(0)$ belonging to the quadratic boundary. This can occur only if the quadratic boundary degenerates to a single feasible point or to affine boundaries. The same arguments can be repeated for $u_k^*(x(k)),\ k=1,\dots,N-1$. □

Remark 1. Theorem 4 relies on a rather strong uniqueness assumption. Sometimes, problem (9)–(11) can be modified in order to obtain uniqueness of the solution and use the result of Theorem 4 which excludes the existence of non-convex ellipsoidal sets. It is reasonable to believe that there are other conditions under which the state-feedback solution is PPWA without claiming uniqueness.

The previous results can be extended to piecewise linear cost functions, i.e., cost functions based on the 1-norm or the ∞-norm.

Theorem 5. *Consider the optimal control problem* (9)–(11) *with* $p=1,\infty$ *and let Assumption* 1 *hold. Then there exists a solution in the form of a PPWA state-feedback control law*

$$u_k^*(x(k))=F_k^i x(k)+G_k^i \quad \text{if } x(k)\in\mathcal{R}_k^i, \tag{23}$$

where $\mathcal{R}_k^i,\ i=1,\dots,N_k^r$, *is a polyhedral partition of the set* $\mathcal{X}_k$ *of feasible states* $x(k)$.

Proof. The proof is similar to the proof of Theorem 3. Fix a certain switching sequence v_i, consider the problem (9)–(11) and constrain the state to switch according to the sequence v_i to obtain problem (16)–(17). Problem (16)–(17) can be viewed as a finite-time optimal control problem with a performance index based on 1-norm or ∞-norm for a linear time-varying system with time-varying constraints and can be solved by using the multiparametric linear program as described in Borrelli (2003). Its solution is a PPWA feedback control law

$$u_0^i(x(0))=\tilde{F}^{i,j}x(0)+\tilde{G}^{i,j},\quad \forall x\in\mathcal{T}^{i,j},\quad j=1,\dots,N^{ri}, \tag{24}$$

and the value function $J_{v_i}^*$ is PWA on polyhedra and convex. The rest of the proof follows the proof of Theorem 3. Note that in this case the value functions to be compared are PWA and not PWQ. □

By comparing Theorems 3 and 5 it is clear that while for performance indices based on 1 or ∞ norms the solution is PWA on polyhedra, in the 2-norm case one may need to deal with non-convex ellipsoidal regions.

7. Computation of the optimal control law via dynamic programming

In the previous section the properties enjoyed by the solution of hybrid optimal control problems were investigated. The proof of Theorem 3 is constructive, but it is based on the enumeration of all the possible switching sequences of the hybrid system, the number of which grows exponentially with the time horizon. Although the computation is performed off-line (the on-line complexity is the one associated with the evaluation of the PWA control law (22)), more efficient methods than enumeration are desirable.

In Bemporad and Morari (1999) the main idea is to translate problem (9)–(11) into a linear or quadratic mixed-integer program that can be solved by using standard commercial software. This approach does not provide the state-feedback law (13) or (23) but only the optimal control sequence $U_N^*(x(0))$ for a given initial state $x(0)$. In Borrelli (2003) the state-feedback law (13) or (23) is computed by means of multiparametric mixed-integer programming. However, the use of multiparametric mixed-integer programming has a major drawback: the solver does not exploit the structure of the optimal control problem. In fact, a large part of the information associated with problem (9)–(11) becomes hidden when it is reformulated as a mixed-integer program. In this section we show how linear and quadratic parametric programming can be used to solve the Hamilton–Jacobi–Bellman equations associated with CFTOC problem (9)–(11). In Baotic, Christophersen, and Morari (2003) we have compared the dynamic programming and the mixed-integer multiparametric programming approach.

The PWA solution (13) will be computed proceeding backwards in time using two tools: a linear or quadratic multiparametric programming solver (depending on the cost function used) and a special technique to store the solution which will be illustrated in the following sections. The algorithm will be presented for optimal control based on a quadratic performance criterion. Its extension to optimal control based on linear performance criteria is straightforward.

7.1. Preliminaries and basic steps

Consider the PWA map ζ defined as

$$\zeta : x \in \mathscr{R}_i \mapsto F_i x + G_i \quad \text{for } i = 1, \ldots, N_{\mathscr{R}}, \tag{25}$$

where $\mathscr{R}_i,\ i = 1, \ldots, N_{\mathscr{R}}$, are subsets of the x-space. Note that if there exist $l, m \in \{1, \ldots, N_{\mathscr{R}}\}$ such that for $x \in \mathscr{R}_l \cap \mathscr{R}_m$, $F_l x + G_l \neq F_m x + G_m$ the map ζ (25) is not single valued.

Definition 8. Given a PWA map (25) we define $f_{\text{PWA}}(x) = \zeta_o(x)$ as the *ordered region single-valued* function associated with (25) when

$$\zeta_o(x) = F_j x + G_j | x \in \mathscr{R}_j \text{ and } \forall i < j : x \notin \mathscr{R}_i,\quad j \in \{1, \ldots, N_{\mathscr{R}}\},$$

and write it in the following form:

$$\zeta_o(x) = \left\downarrow \begin{array}{ll} F_1 x + G_1 & \text{if } x \in \mathscr{R}_1, \\ & \vdots \\ F_{N_{\mathscr{R}}} x + G_{N_{\mathscr{R}}} & \text{if } x \in \mathscr{R}_{N_{\mathscr{R}}}. \end{array} \right.$$

Note that given a PWA map (25) the corresponding *ordered region single-valued* function ζ_o changes if the order used to store the regions $\mathscr{R}_i$ and the corresponding affine gains change. For illustration purposes consider the example

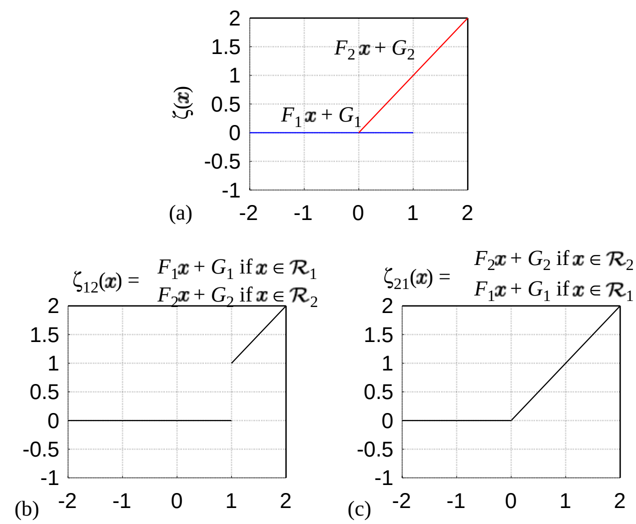

Fig. 2. Illustration of the ordered region single-valued function: (a) Multi valued PWA map ζ; (b) Ordered region single valued function ζ_{12}; (c) Ordered region single valued function ζ_{21}.

depicted in Fig. 2, where $x \in \mathbb{R}$, $N_{\mathscr{R}} = 2$, $F_1 = 0$, $G_1 = 0$, $\mathscr{R}_1 = [-2, 1]$, $F_2 = 1$, $G_2 = 0$, $\mathscr{R}_2 = [0, 2]$.

In the following we assume that the sets $\mathscr{R}_i^k$ in the optimal solution (13) can overlap. When we refer to the PWA function $u_k^*(x(k))$ in (13) we will implicitly mean the ordered region single-valued function associated with the mapping (13).

Example 7.1. Let $J_1^* : \mathscr{R}_1 \to \mathbb{R}$ and $J_2^* : \mathscr{R}_2 \to \mathbb{R}$ be two quadratic functions, $J_1^*(x) \triangleq x' L_1 x + M_1 x + N_1$ and $J_2^*(x) \triangleq x' L_2 x + M_2 x + N_2$, where $\mathscr{R}_1$ and $\mathscr{R}_2$ are convex polyhedra and $J_i^*(x) = +\infty$ if $x \notin \mathscr{R}_i$, $i \in \{1, 2\}$. Let $u_1^* : \mathscr{R}_1 \to \mathbb{R}^m$, $u_2^* : \mathscr{R}_2 \to \mathbb{R}^m$ be vector functions. Let $\mathscr{R}_1 \cap \mathscr{R}_2 \triangleq \mathscr{R}_3 \neq \emptyset$ and define

$$J^*(x) \triangleq \min\{J_1^*(x), J_2^*(x)\}, \tag{26}$$

$$u^*(x) \triangleq \begin{cases} u_1^*(x) & \text{if } J_1^*(x) \leqslant J_2^*(x), \\ u_2^*(x) & \text{if } J_1^*(x) \geqslant J_2^*(x), \end{cases} \tag{27}$$

where $u^*(x)$ can be a set valued function. Let $L_3 = L_2 - L_1$, $M_3 = M_2 - M_1$, $N_3 = N_2 - N_1$. Then, corresponding to the three following cases:

(i) $J_1^*(x) \leqslant J_2^*(x)\ \forall x \in \mathscr{R}_3$,
(ii) $J_1^*(x) \geqslant J_2^*(x)\ \forall x \in \mathscr{R}_3$,
(iii) $\exists x_1, x_2 \in \mathscr{R}_3 | J_1^*(x_1) < J_2^*(x_1)$ and $J_1^*(x_2) > J_2^*(x_2)$,

the expressions (26) and a real-valued function that can be extracted from (27) can be written equivalently as:

(1)

$$J^*(x) = \left\downarrow \begin{array}{ll} J_1^*(x) & \text{if } x \in \mathscr{R}_1, \\ J_2^*(x) & \text{if } x \in \mathscr{R}_2, \end{array} \right. \tag{28}$$

$$u^*(x) = \begin{cases} u_1^*(x) & \text{if } x \in \mathscr{R}_1, \\ u_2^*(x) & \text{if } x \in \mathscr{R}_2. \end{cases} \quad (29)$$

(2) As in (28) and (29) by switching the indices 1 and 2.
(3)

$$J^*(x) = \begin{cases} \min\{J_1^*(x), J_2^*(x)\} & \text{if } x \in \mathscr{R}_3, \\ J_1^*(x) & \text{if } x \in \mathscr{R}_1, \\ J_2^*(x) & \text{if } x \in \mathscr{R}_2, \end{cases} \quad (30)$$

$$u^*(x) = \begin{cases} u_1^*(x) & \text{if } x \in \mathscr{R}_3 \cap \{x| \\ & \quad x'L_3x + M_3x + N_3 \geqslant 0\}, \\ u_2^*(x) & \text{if } x \in \mathscr{R}_3 \cap \{x| \\ & \quad x'L_3x + M_3x + N_3 \leqslant 0\}, \\ u_1^*(x) & \text{if } x \in \mathscr{R}_1, \\ u_2^*(x) & \text{if } x \in \mathscr{R}_2, \end{cases} \quad (31)$$

where (28)–(31) have to be considered as PWA and PPWQ functions in the *ordered region* sense.

Example 7.1 shows how to

- avoid the storage of the intersections of two polyhedra in cases (i) and (ii);
- avoid the storage of possibly non-convex regions $\mathscr{R}_1 \backslash \mathscr{R}_3$ and $\mathscr{R}_2 \backslash \mathscr{R}_3$;
- work with multiple quadratic functions instead of quadratic functions defined over non-convex and non-polyhedral regions.

The three points listed above will be the three basic ingredients for storing and simplifying the optimal control law (13). Next we will show how to compute it.

Remark 2. To distinguish between cases (i), (ii) and (iii) of Example 7.1, in general, one needs to solve an indefinite quadratic program, namely,

$$\begin{aligned} &\min_x \quad x'L_3x + M_3x + N_3 \\ &\text{subj. to} \quad x \in \mathscr{R}_3. \end{aligned} \quad (32)$$

In our approach, to avoid such a test form (31) corresponding to case (iii) can be used. The only drawback is that form (31) is, in general, a non-minimal representation of the value function and therefore it increases the complexity of evaluating and storing the optimal control profile (13).

7.2. Multiparametric programming with multiple quadratic functions

Consider the multiparametric program

$$\begin{aligned} J^*(x) \triangleq \min_u \quad & l(x,u) + q(f(x,u)) \\ \text{s.t.} \quad & f(x,u) \in \mathscr{R}, \end{aligned} \quad (33)$$

where $\mathscr{R} \subseteq \mathbb{R}^n$ is a compact set, $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$, $q : \mathscr{R} \to \mathbb{R}$, and $l : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is a convex quadratic function of x and u. We aim at determining the region $\mathscr{X}$ of variables x such that program (33) is feasible and the optimum $J^*(x)$ is finite, and at finding the expression $u^*(x)$ of (one of) the optimizer(s). We point out that the constraint $f(x,u) \in \mathscr{R}$ implies a constraint on u as a function of x since u can assume only values where $f(x,u)$ is defined.

Next we show how to solve several forms of problem (33).

Lemma 1 (*one to one problem*). *Problem* (33) *where f is linear, q is quadratic and strictly convex, and $\mathscr{R}$ is a polyhedron can be solved by one mp-QP.*

Proof. See Bemporad et al. (2002). □

Lemma 2 (*one to one problem of multiplicity d*). *Problem* (33) *where f is linear, q is a multiple quadratic function of multiplicity d and $\mathscr{R}$ is a polyhedron can be solved by d mp-QPs.*

Proof. The multiparametric program to be solved is

$$\begin{aligned} J^*(x) = \min_u \quad & \{l(x,u) + \\ & \min\{q_1(f(x,u)), \ldots, q_d(f(x,u))\}\} \\ \text{subj. to} \quad & f(x,u) \in \mathscr{R}, \end{aligned} \quad (34)$$

and it is equivalent to

$$J^*(x) = \min \left\{ \begin{array}{l} \min_u l(x,u) + q_1(f(x,u)), \\ \text{subj. to } f(x,u) \in \mathscr{R}, \\ \vdots \\ \min_u l(x,u) + q_d(f(x,u))\} \\ \text{subj. to } f(x,u) \in \mathscr{R} \end{array} \right\}. \quad (35)$$

The ith sub-problems in (35)

$$J_i^*(x) \triangleq \min_u l(x,u) + q_i(f(x,u)) \quad (36)$$

$$\text{subj. to } f(x,u) \in \mathscr{R} \quad (37)$$

is a *one to one problem* and therefore it is solvable by an mp-QP. Let the solution of the ith mp-QPs be

$$u^i(x) = \tilde{F}^{i,j}x + \tilde{G}^{i,j}, \quad \forall x \in \mathscr{T}^{i,j}, \quad j = 1, \ldots, N^{ri}, \quad (38)$$

where $\mathscr{T}^i = \bigcup_{j=1}^{N^{ri}} \mathscr{T}^{i,j}$ is a polyhedral partition of the convex set $\mathscr{T}^i$ of feasible x for the ith sub-problem and N^{ri} is the corresponding number of polyhedral regions. The feasible set $\mathscr{X}$ satisfies $\mathscr{X} = \mathscr{T}^1 = \cdots = \mathscr{T}^d$ since the constraints of the d sub-problems are identical.

The solution $u^*(x)$ to the original problem (34) is obtained by comparing and storing the solution of d mp-QP sub-problems (36)–(37) as explained in Example 7.1.

Consider the case $d=2$, and consider the intersection of the polyhedra $\mathscr{T}^{1,i}$ and $\mathscr{T}^{2,l}$ for $i=1,\ldots,N^{r1}$, $l=1,\ldots,N^{r2}$. For all $\mathscr{T}^{1,i}\cap\mathscr{T}^{2,l}\triangleq\mathscr{T}^{(1,i),(2,l)}\neq\emptyset$ the optimal solution is stored in an ordered way as described in Example 7.1, while paying attention to the fact that a region could be already stored. Moreover, when storing a new polyhedron with the corresponding value function and optimizer, the relative order of the regions already stored must not be changed. The result of this *intersect and compare* procedure is

$$u^*(x)=F^ix+G^i \quad \text{if } x\in\mathscr{R}^i,$$
$$\mathscr{R}^i\triangleq\{x:\ x'L^i(j)x+M^i(j)x\leqslant N^i(j),\ \ j=1,\ldots,n^i\}, \tag{39}$$

where $\mathscr{R}=\bigcup_{j=1}^{N_{\mathscr{R}}}\mathscr{R}^j$ is a polyhedron and the value function

$$J^*(x)=\tilde{J}_j^*(x) \quad \text{if } x\in\mathscr{D}^j,\ \ j=1,\ldots,N^{\mathscr{D}}, \tag{40}$$

where $\tilde{J}_j^*(x)$ are multiple quadratic functions defined over the convex polyhedra $\mathscr{D}^j$. The polyhedra $\mathscr{D}^j$ can contain several regions $\mathscr{R}^i$ or can coincide with one of them. Note that (39) and (40) have to be considered as PWA and PPWQ functions in the *ordered region* sense.

If $d>2$ then the value function in (40) is intersected with the solution of the third mp-QP sub-problem and the procedure is iterated by making sure not to change the relative order of the polyhedra and corresponding gain of the solution constructed in the previous steps. The solution will still have the same form (39)–(40). □

Lemma 3 (*one to r problem*). *Problem* (33) *where f is linear, q is a lower-semicontinuous PPWQ function defined over r polyhedral regions and strictly convex on each polyhedron, and $\mathscr{R}$ is a polyhedron, can be solved by r mp-QPs.*

Proof. Let $q(x)\triangleq q_i$, if $x\in\mathscr{R}_i$, be the PWQ function where the closures $\bar{\mathscr{R}}_i$ of $\mathscr{R}_i$ are polyhedra and q_i strictly convex quadratic functions. The multiparametric program to solve is

$$J^*(x)=\min\left\{\begin{array}{l}\min\limits_u l(x,u)+q_1(f(x,u)),\\ \text{subj. to } f(x,u)\in\bar{\mathscr{R}}_1\\ \qquad\qquad f(x,u)\in\mathscr{R}\\ \vdots\\ \min\limits_u l(x,u)+q_r(f(x,u))\}\\ \text{subj. to } f(x,u)\in\bar{\mathscr{R}}_r\\ \qquad\qquad f(x,u)\in\mathscr{R}\end{array}\right\}. \tag{41}$$

The proof follows the lines to the proof of the previous theorem with the exception that the constraints of the *i*th mp-QP sub-problem differ from the one of the *j*th mp-QP sub-problem, $i\neq j$.

The lower-semicontinuity assumption on $q(x)$ allows one to use the closure of the sets $\mathscr{R}_i$ in (41). The cost function in problem (33) is lower-semicontinuous since it is a composition of a lower-semicontinuous function and a continuous function. Then, since the domain is compact, problem (41) admits a minimum. Therefore, for a given x, there exists one mp-QP in problem (41) which yields the optimal solution. There might exist other mp-QP solutions in (41) feasible at x that are neither optimal nor feasible for the original problem (33). However, since $q(x)$ is lower-semicontinuous, such solutions will be discarded when the corresponding value functions are compared. The procedure based on solving mp-QPs and storing the results as in Example 7.1 will be the same as in Lemma 2 but the domain $\mathscr{R}=\bigcup_{j=1}^{N^{\mathscr{R}}}\mathscr{R}^j$ of the solution can be a non-Euclidean polyhedron. □

If f is PPWA and defined over s regions then we have an *s to X problem* where *X* can belong to any of the problems listed above. In particular, we have an *s to r problem of multiplicity d* if f is PPWA and defined over s regions and q is a multiple PPWQ function of multiplicity d, defined over r polyhedral regions. The following lemma can be proven along the lines of the proofs given before.

Lemma 4. *Problem* (33) *where f is linear and q is a lower-semicontinuous PPWQ function of multiplicity d, defined over r polyhedral regions and strictly convex on each polyhedron, is a one to r problem of multiplicity d and can be solved by $r\cdot d$ mp-QPs.*

An s to *r* problem of multiplicity *d can be decomposed into s* one to *r* problems of multiplicity *d*. *An s* to one problem *can be decomposed into s* one to one problems.

7.3. Algorithmic solution of the HJB equations

In the following we will substitute the CPWA system equations (8) with the shorter form

$$x(k+1)=\tilde{f}_{\mathrm{PWA}}(x(k),u(k)), \tag{42}$$

where $\tilde{f}_{\mathrm{PWA}}:\tilde{\mathscr{P}}\to\mathbb{R}^n$ and $\tilde{f}_{\mathrm{PWA}}(x,u)=A^ix+B^iu+f^i$ if $[\begin{smallmatrix}x\\u\end{smallmatrix}]\in\tilde{\mathscr{P}}^i$, $i=1,\ldots,s$, and $\{\tilde{\mathscr{P}}^i\}$ is a polyhedral partition of $\tilde{\mathscr{P}}$.

Consider the dynamic programming formulation of the CFTOC problem (9)–(11),

$$J_j^*(x(j))\triangleq\min_{u_j}\ \|Qx_j\|_2+\|Ru_j\|_2$$
$$+J_{j+1}^*(\tilde{f}_{\mathrm{PWA}}(x(j),u_j)) \tag{43}$$
$$\text{subj. to } \tilde{f}_{\mathrm{PWA}}(x(j),u_j)\in\mathscr{X}_{j+1} \tag{44}$$

for $j = N-1, \ldots, 0$, with terminal conditions

$$\mathscr{X}_N = \mathscr{X}_f \tag{45}$$

$$J_N^*(x) = \|Px\|_2, \tag{46}$$

where $\mathscr{X}_j$ is the set of all states $x(j)$ for which problem (43)–(44) is feasible:

$$\mathscr{X}_j = \{x \in \mathbb{R}^n | \exists u,\ \tilde{f}_{\mathrm{PWA}}(x,u) \in \mathscr{X}_{j+1}\}. \tag{47}$$

Eqs. (43)–(47) are the discrete-time version of the well-known Hamilton–Jacobi–Bellman equations for continuous-time optimal control problems.

Assume for the moment that there are no binary inputs and binary states, $m_\ell = n_\ell = 0$. The HJB equations (43)–(46) can be solved backwards in time by using a multiparametric quadratic programming solver and the results of the previous section.

Consider the first step of the dynamic program (43)–(46)

$$J_{N-1}^*(x_{N-1}) \triangleq \min_{\{u_{N-1}\}} \|Qx_{N-1}\|_2 + \|Ru_{N-1}\|_2 + J_N^*(\tilde{f}_{\mathrm{PWA}}(x_{N-1}, u_{N-1})), \tag{48}$$

$$\text{subj. to } \tilde{f}_{\mathrm{PWA}}(x_{N-1}, u_{N-1}) \in \mathscr{X}_f. \tag{49}$$

The cost to go function $J_N^*(x)$ in (48) is quadratic, the terminal region $\mathscr{X}_f$ is a polyhedron and the constraints are PWA. problem (48)–(49) is an *s to one problem* that can be solved by solving s mp-QPs (Lemma 4). From the second step $j = N-2$ to the last one $j = 0$ the cost to go function $J_{j+1}^*(x)$ is a lower-semicontinuous PPWQ with a certain multiplicity d_{j+1}, the terminal region $\mathscr{X}_{j+1}$ is a polyhedron (in general non-Euclidean) and the constraints are PWA. Therefore, problem (43)–(46) is an s to N_{j+1}^r *problem with multiplicity* d_{j+1} (where N_{j+1}^r is the number of polyhedra of the cost to go function J_{j+1}^*), which can be solved by solving $sN_{j+1}^r d_{j+1}$ mp-QPs (Lemma 4). The resulting optimal solution will have form (13) considered in the ordered region sense.

In the presence of binary inputs the procedure can be repeated, with the difference that all the possible combinations of binary inputs must be enumerated. Therefore, a *one to one problem* becomes a 2^{m_ℓ} *to one problem* and so on. In the presence of binary states the procedure can be repeated either by enumerating them all or by solving a dynamic programming algorithm at time step k from a relaxed state space to the set of binary states feasible at time $k+1$.

Next we summarize the main steps of the dynamic programming algorithm discussed in this section. We use boldface characters to denote sets of polyhedra, i.e., $\mathbf{R} := \{\mathscr{R}_i\}_{i=1,\ldots,|\mathbf{R}|}$, where $\mathscr{R}_i$ is a polyhedron and $|\mathbf{R}|$ is the cardinality of the set $\mathbf{R}$. Furthermore, when we say SOLVE an mp-QP we mean to compute and store the triplet $S_{k,i,j}$ of expressions for the value function, the optimizer and the polyhedral partition of the feasible space.

Algorithm 7.1.

Input: CFTOC problem (9)–(11)
Output: Solution (13) in the ordered region sense.

1 **let** $\mathbf{R}_N = \{\mathscr{X}_f\}$
2 **let** $J_{N,1}^*(x) := x'Px$
3 **for** $k = N-1, \ldots, 1$,
4 **for** $i = 1, \ldots, |\mathbf{R}_{k+1}|$,
5 **for** $j = 1, \ldots, s$,
6 **let** $\mathscr{S}_{k,i,j} = \{\}$
7 SOLVE the mp-QP

$$S_{k,i,j} \leftarrow \min_{u_k} x_k' Q x_k + u_k' R u_k + J_{k+1,i}^*(A_j x_k + B_j u_k + f_j)$$
$$\text{subj. to } \begin{cases} A_j x_k + B_j u_k + f_j \in \mathscr{R}_{k+1,i} \\ \begin{bmatrix} x_k \\ u_k \end{bmatrix} \in \tilde{\mathscr{P}}^j \end{cases}$$

8 **end**
9 **end**
10 Let $\mathbf{R}_k = \{\mathscr{R}_{k,i,j,l}\}_{i,j,l}$. Denote by $\mathscr{R}_{k,h}$ its elements, and by $J_{k,h}^*$ and $u_{k,h}^*(x)$ the associated costs and optimizers, with $h \in \{1, \ldots, |\mathbf{R}_k|\}$
11 KEEP only triplets $(J_{k,h}^*(x), u_{k,h}^*(x), \mathscr{R}_{k,h})$ for which
$\exists x \in \mathscr{R}_{k,h} : x \notin \mathscr{R}_{k,d}, \forall d \neq h$ OR
$\exists x \in \mathscr{R}_{k,h} : J_{k,h}^*(x) < J_{k,d}^*(x), \forall d \neq h$
12 CREATE multiplicity information and additional regions for an ordered region solution as explained in Example 7.1
13 **end**.

In Algorithm 7.1, the structure $S_{k,i,j}$ stores the matrices defining quadratic function $J_{k,i,j,l}^*(\cdot)$, affine function $u_{k,i,j,l}^*(\cdot)$ and polyhedra $\mathscr{R}_{k,i,j,l}$, for all l:

$$S_{k,i,j} = \bigcup_l \{(J_{k,i,j,l}^*(x), u_{k,i,j,l}^*(x), \mathscr{R}_{k,i,j,l})\}, \tag{50}$$

where the indices in (50) have the following meaning: k is the time step, i indexes the piece of the "cost-to-go" function that the DP algorithm is considering, j indexes the piece of the PWA dynamics the DP algorithm is considering, and l indexes the polyhedron in the mp-QP solution of the (k, i, j)th mp-QP problem.

Step 11 of Algorithm 7.1 aims at discarding regions $\mathscr{R}_{k,h}$ that are completely covered by some other regions that have lower cost. Obviously, if there are some parts of the region $\mathscr{R}_{k,h}$ that are not covered at all by other regions (first condition) we need to keep it. Note that comparing the cost functions (second condition) is, in general, non-convex optimization problem. One might consider solving the problem exactly, but since the algorithm works even if some removable regions are kept, we usually formulate LMI relaxation of the problem at hand. While executing Step 11 of Algorithm 7.1 we can simultaneously obtain the information of multiplicity of polyhedral subsets of the region $\mathscr{R}_{k,h}$.

The output of Algorithm 7.1 is the state-feedback control law (13) considered in the ordered region sense. The online implementation of the control law requires simply the evaluation of the PWA controller (13) in the ordered region sense (note that the order the solution is stored is important).

8. Discontinuous PWA systems

Without Assumption 1 the optimal control problem (9)–(11) may be feasible but may not admit an optimizer for some $x(0)$ (the problem in this case should be to find an infimum rather than the minimum).

Under the assumption that the optimizer exists for all states $x(k)$, the approach explained in the previous sections can be applied to discontinuous systems by considering three elements. First, the PWA system (8) has to be defined on each polyhedron of its domain *and all its lower dimensional facets*. Secondly, dynamic programming has to be performed "from" and "to" any lower dimensional facet of each polyhedron of the PWA domain. Finally, value functions are not lower-semicontinuous, which implies that Lemma 3 cannot by used. Therefore, when considering the closure of polyhedral domains in multiparametric programming (41), a post-processing is necessary in order to remove multiparametric optimal solutions which do not belong to the original set but only to its closure. The tedious details of the dynamic programming algorithm for discontinuous PWA systems are not included in this paper but can be immediately extracted from the results of the previous sections.

In practice, the approach just described for discontinuous PWA systems can easily be numerically prohibitive. The simplest approach from a practical point of view resorts to introducing gaps between the boundaries of any two polyhedra belonging to the PWA domain (or, equivalently, to shrinking by a quantity ε the size of every polyhedron of the original PWA system). In this way, one deals with PWA systems defined over a disconnected union of closed polyhedra. By doing so, one can use the approach discussed previously in this paper for continuous PWA systems. However, the optimal controller will not be defined at the points in the gaps and at points when the only feasible solution is in the gaps. Also, the computed solution might be arbitrarily different from the original solution to problem (9)–(11) at any feasible point x. Despite this, if the dimension ε of the gaps is close to the machine precision and comparable to sensor/estimation errors, such an approach is very appealing in practice. To the best of our knowledge in some cases this approach is the only computationally tractable for computing controllers for discontinuous hybrid systems fulfilling state and input constraints that are implementable in real-time.

Without Assumption 1, problem (9)–(11) is well defined only if an optimizer exists for all $x(0)$. In general, this is not easy to check. The dynamic programming algorithm described in this paper could be used for such test but the details are not included in this paper.

9. Conclusions

For discrete-time linear hybrid systems, we have described an off-line procedure to synthesize optimal control laws based on the minimization of quadratic and linear performance indices subject to linear constraints on inputs and states. The procedure is based on a combination of dynamic programming and multiparametric quadratic programming. In collaboration with different companies and institutes, the results described in this paper have been applied to a wide range of problems (Baotic et al., 2003; Bemporad et al., 2002, 2003; Bemporad & Morari, 1999; Borrelli et al., 2001; Ferrari-Trecate et al., 2002; Möbus et al., 2003; Torrisi & Bemporad, 2004). Simple examples that highlight the main features of the hybrid system approach presented in this paper can be found in Borrelli et al. (2003).

Acknowledgements

This work was partially supported by the Swiss National Science Foundation and the European Project "Computation and Control". We thank the anonymous reviewers, David Q. Mayne, Christos G. Cassandras, Nicolo' Giorgetti and Stefano Di Cairano for constructive and helpful comments on the original manuscript.

References

Antsaklis, P. J. (2000). A brief introduction to the theory and applications of hybrid systems. *Proceedings of the IEEE, Special Issue on Hybrid Systems: Theory and Applications*, *88*(7), 879–886.

Baotic, M., Christophersen, F.J., & Morari, M. (2003). A new algorithm for constrained finite time optimal control of hybrid systems with a linear performance index. In *Proceedings of European control conference*, University of Cambridge, UK.

Baotic, M., Vasak, M., Morari, M., & Peric, N. (2003). Hybrid theory based optimal control of electronic throttle. In *Proceedings of the American control conference*, Denver, Colorado, USA.

Bemporad, A. (2004). Efficient conversion of mixed logical dynamical systems into an equivalent piecewise affine form. *IEEE Transactions on Automatic Control*, *49*(5), 832–838.

Bemporad, A., Borodani, P., & Mannelli, M. (2003). Hybrid control of an automotive robotized gearbox for reduction of consumptions and emissions. In O. Maler, & A. Pnueli (Eds.), *Hybrid systems: Computation and control, HSCC 2003, Lecture notes in computer science* (Vol. 2623, pp. 81–96). Berlin: Springer.

Bemporad, A., Giorgetti, N., Kolmanovsky, I. V., & Hrovat, D. (2002). Hybrid modeling and control of a direct injection stratified charge engine. In *Symposium on advanced automotive technologies, ASME international mechanical engineering congress and exposition*, New Orleans, Louisiana.

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

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

Berge, C. (1997). *Topological spaces*. Mineola New York: Dover Publications.

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

Borrelli, F., Baotic, M., Bemporad, A., & Morari, M. (2003). *Constrained optimal control of discrete-time linear hybrid systems*. Technical Report AUT03-05, Automatic Control Laboratory, ETH Zurich, Switzerland.

Borrelli, F., Bemporad, A., Fodor, M., & Hrovat, D. (2001). A hybrid approach to traction control. In A. Sangiovanni-Vincentelli, & M. D. Di Benedetto (Eds.), *Hybrid systems: Computation and control*, *Lecture notes in computer science* (Vol. 2034 pp. 162–174). Berlin: Springer.

Branicky, M. S., Borkar, V. S., & Mitter, S. K. (1998). A unified framework for hybrid control: Model and optimal control theory. *IEEE Transactions on Automatic Control*, *43*(1), 31–45.

Ferrari-Trecate, G., Gallestey, E., Stothert, A., Letizia, P., Spedicato, M., Morari, et al. (2002). Modeling and control of co-generation power plants: A hybrid system approach. In C. J. Tomlin, & M. R. Greenstreet (Eds.), *Proceedings of the fifth international workshop on hybrid systems: Computation and control*, *Lecture notes in computer science*. Berlin: Springer.

Fiacco, A. V. (1976). Sensitivity analysis for nonlinear programming using penalty methods. *Mathematical Programming*, *10*(3), 287–311.

Fiacco, A. V. (1983). *Introduction to sensitivity and stability analysis in nonlinear programming*. London, UK: Academic Press.

Heemels, W. P. M. H., De Schutter, B., & Bemporad, A. (2001). Equivalence of hybrid dynamical models. *Automatica*, *37*(7), 1085–1091.

Hogan, W. M. (1973). Point-to-set maps in mathematical programming. *SIAM Review*, *15*(3), 591–603.

Lygeros, J., Tomlin, C., & Sastry, S. (1999). Controllers for reachability specifications for hybrid systems. *Automatica*, *35*(3), 349–370.

Mayne, D. Q. (2001). Constrained optimal control. In *European control conference*, *Plenary lecture*, Porto, Portugal.

Mignone, D. (2002). *Control and estimation of hybrid systems via mathematical optimization*. Dr. sc. tech. Thesis, Automatic Control Laboratory - ETH, Zurich, 2002. http://control.ee.ethz.ch.

Möbus, R., Baotic, M., & Morari, M. (2003). Multi-object adaptive cruise control. In O. Maler, & A. Pnueli (Eds.), *Hybrid systems: Computation and control, HSCC 2003*, *Lecture notes in computer science* (Vol. 2623 pp. 359–374). Berlin: Springer.

Sontag, E. D. (1981). Nonlinear regulation: The piecewise linear approach. *IEEE Transactions on Automatic Control*, *26*(2), 346–358.

Torrisi, F. D., & Bemporad, A. (2004). HYSDEL—A tool for generating computational hybrid models. *IEEE Transactions on Control Systems Technology*, *12*(2), 235–249.

Francesco Borrelli received the "Laurea" degree in computer science engineering in 1998 from the University of Naples "Federico II", Italy. In 2002 he received the Ph.D. from the Automatic Control Laboratory at ETH-Zurich, Switzerland. In 2003 he received the ETH Medal for the best Ph.D. dissertation. He has been a research assistant at the Automatic Control Laboratory of the ETH-Zurich and a contract assistant professor at the Aerospace and Mechanics Department at the University of Minnesota, USA. Currently he is an assistant professor at the "Universitá del Sannio", Benevento, Italy. He is author of the book "Constrained Optimal Control of Linear and Hybrid Systems" published by Springer Verlag. He is the winner of the "Innovation Prize 2004" from the ElectroSwiss Foundation. His research interests include constrained optimal control, model predictive control, robust control, parametric programming, singularly perturbed systems and automotive applications of automatic control.

Mato Baotić received the B.Sc. and M.Sc. degrees, both in Electrical Engineering, from the Faculty of Electrical Engineering and Computing (FEEC), University of Zagreb, Croatia, in 1997 and 2000, respectively. He received a scholarship of the Swiss Government and spent a school year 2000/01 as a visiting researcher at the Automatic Control Lab, Swiss Federal Institute of Technology (ETH) Zurich. In 2005 he received the Ph.D. from the ETH Zurich, Switzerland. Currently he is a post-doc at the Department of Control and Computer Engineering in Automation, FEEC, University of Zagreb, Croatia. His research interests include mathematical programming, hybrid systems, optimal control and model predictive control.

Alberto Bemporad received the master degree in Electrical Engineering in 1993 and the Ph.D. in Control Engineering in 1997 from the University of Florence, Italy. He spent the academic year 1996/97 at the Center for Robotics and Automation, Dept. Systems Science & Mathematics, Washington University, St. Louis, as a visiting researcher. In 1997–1999, he held a postdoctoral position at the Automatic Control Lab, ETH, Zurich, Switzerland, where he collaborated as a senior researcher in 2000–2002. Since 1999, he is assistant professor at the University of Siena, Italy. He has published papers in the area of hybrid systems, model predictive control, multiparametric optimization, computational geometry, robotics and automotive control. He is coauthor of the Model Predictive Control Toolbox (The Mathworks, Inc.) and author of the Hybrid Toolbox. He is an Associate Editor of the IEEE Transactions on Automatic Control, and Chair of the IEEE Control Systems Society Technical Committee on Hybrid Systems.

Manfred Morari was appointed head of the Automatic Control Laboratory at the Swiss Federal Institute of Technology (ETH) in Zurich in 1994. Before that he was the McCollum-Corcoran Professor of Chemical Engineering and Executive Officer for Control and Dynamical Systems at the California Institute of Technology. He obtained the diploma from ETH Zurich and the Ph.D. from the University of Minnesota, both in chemical engineering. His interests are in hybrid systems and the control of biomedical systems. In recognition of his research contributions, he received numerous awards, among them the Donald P. Eckman Award of the Automatic Control Council, the Allan P. Colburn Award and the Professional Progress Award of the AIChE, the Curtis W. McGraw Research Award of the ASEE, Doctor Honoris Causa from Babes-Bolyai University, Fellow of IEEE, the IEEE Control Systems Award, and was elected to the National Academy of Engineering (US). Professor Morari has held appointments with Exxon and ICI plc and serves on the technical advisory board of several major corporations.