FEASIBLE MODE ENUMERATION AND COST COMPARISON FOR EXPLICIT QUADRATIC MODEL PREDICTIVE CONTROL OF HYBRID SYSTEMS

Alessandro Alessio Alberto Bemporad

Dip. Ingegneria dell'Informazione, University of Siena, Italy. Email: `alessio,bemporad@dii.unisi.it.`

Abstract For hybrid systems in piecewise affine (PWA) form, this paper presents a new methodology for computing the solution, defined over a set of (possibly overlapping) polyhedra, of the finite-time constrained optimal control problem based on quadratic costs. First, feasible mode sequences are determined via backward reachability analysis, and multiparametric quadratic programming is employed to determine candidate polyhedral regions of the solution and the corresponding value functions and optimal control gains. Then, the value functions associated with overlapping regions are compared in order to discard those regions whose associated control law is never optimal. The comparison problem is, in general, nonconvex and is tackled here as a DC (Difference of Convex functions) programming problem.

1. INTRODUCTION

In the recent years, different methods for the design and the analysis of controllers for hybrid systems have been studied (see e.g. (Corona, 2005) and references therein). In particular, multiparametric programming techniques were proposed to synthesize state-feedback controllers defined over a set of polyhedral regions, by solving a finite-time optimal control problems explicitly with respect to the state and reference vectors.

(Bemporad *et al.*, 2000) proposed a procedure for synthesizing piecewise affine optimal controllers for discrete-time linear hybrid systems. A state feedback solution of a finite-time optimal control problem with performance criteria based on linear (1 or ∞) norms is obtained using multiparametric mixed-integer linear programming. A different approach based on dynamic programming was proposed in (Baotic *et al.*, 2003). The use of linear norms has some practical disadvantages, due to the fact that typically good performance can only be achieved with long time horizons. Moreover, the resulting state-space partition is typically very complex, because of the large number of regions.

Quadratic costs allow one to achieve better performances with shorter horizons, although the partition associated with the fully explicit optimal solution to a finite time constrained optimal control (FTCOC) problem for hybrid systems may not be polyhedral (Borrelli *et al.*, 2005).

(Borrelli *et al.*, 2005) proposed an algorithm for computing the solution to the FTCOC problem with quadratic costs. The procedure is based on dynamic programming (DP) iterations. Multiparametric quadratic programs (mpQP) (Bemporad *et al.*, 2002) are solved at each iteration, and quadratic value functions are compared to possibly eliminate regions that are proved to never be optimal. In typical situations the total number of solved mpQPs (as well as of generated polyhedral regions) grows exponentially, and suffers the drawback of an excessive partitioning of the state space.

A different approach was proposed in (Mayne, 2001; Mayne and Rakovic, 2002), where the authors propose to enumerate all possible switching sequences, and for each sequence convert the PWA dynamics into a time-varying system and solve an optimal control problem explicitly via mpQP. As any given initial state may lie in more than one polyhedral region, the associated control gain giving the smallest cost needs to be selected by on-line comparison. This leads to an exponential number of mpQPs that need to be solved and a possibly large *on-line* CPU time spent for comparing the cost functions.

In this paper we propose a different approach that exploits dynamic programming ideas (more precisely, backwards reachability analysis) to obtain all the feasible mode sequences (therefore avoiding an explicit enumeration of all of them), and that, after solving an mpQP for each sequence, post-processes the resulting polyhedral partitions to eliminate all the regions (and their associated control gains) that never provide the lowest cost, using a novel DC (Difference of Convex functions) algorithm. The resulting number of total regions that needs to be stored is minimized, and therefore the CPU time needed by the on-line procedure for searching the region with minimum cost is reduced.

2. HYBRID MPC SETUP

Consider the *Piecewise Affine System* (PWA) described by the relations

$$\begin{array}{ll} x(k+1) = A_i x(k) + B_i u(k) + f_i, & \text{if } \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} \in \mathcal{X}_i \\ y(k) \quad\;\; = C_i x(k) + g_i & \end{array} \quad (1)$$

where $\{\mathcal{X}_i\}_{i=1}^s$ is a polyhedral partition of the state+input set. Suppose there are no binary states and inputs so that $x(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^m$, and A_i, B_i, f_i, C_i, g_i are matrices of suitable dimension [1]. Hybrid systems of the form (1) can be obtained for instance by system identification tools or by converting HYSDEL models using the method of (Bemporad, 2004).

Model Predictive Control (MPC) ideas can be applied to control hybrid models of the form (1). Here, at each sampling time, an open-loop optimal control problem is solved over a finite horizon N. Only the first sample of the optimal sequence is then applied to the plant at time k. At the next time step, a new optimal control problem based on new measurements of the state is solved over a shifted horizon. The solution relies on the hybrid model (1) of the system dynamics, minimizes a performance figure, and respects all input, output and state constraints.

For simplicity of notation, assume that we want to regulate the system state to the origin. So, the MPC open-loop optimal control problem can be formulated as follows

$$V^*(x(0)) = \min_U J(U, x(0))$$

$$J(U, x(0)) = x'(N)Px(N) + \sum_{k=0}^{N-1} x'(k)Qx(k) + u'(k)Ru(k) \quad (2a)$$

$$\begin{aligned} \text{s.t. } & \text{PWA Model (1)} \\ & x_{\min} \le x(k) \le x_{\max}, \\ & y_{\min} \le y(k) \le y_{\max}, \ k = 0, \ldots, N-1, \\ & u_{\min} \le u(k) \le u_{\max}, \\ & x(N) \in \mathcal{X}^N, \end{aligned} \quad (2b)$$

where N is the control horizon, and $U \triangleq \{u(k),\ u(k+1),\ u(k+N-1)\}$ is the input sequence to be optimized. The bounds $u_{\min}$, $u_{\max}$, $x_{\min}$, $x_{\max}$, $y_{\min}$, $y_{\max}$ impose limits on inputs, states, and outputs, respectively, P is a weight on the terminal state, and $\mathcal{X}^N$ is a terminal set contained in the box $\{x : x_{\min} \le x \le x_{\max}\}$.

3. EXPLICIT SOLUTION FOR A FIXED MODE SEQUENCE

Problem (2) is usually referred to as the Finite Time Constrained Optimal Control (FTCOC) based on quadratic costs (Borrelli *et al.*, 2005). With an MPC synthesis in mind, our goal is to find the first optimal move $u^*(x(0))$ as a function of the initial state $x(0)$. While for *a given* $x(0)$ the input $u^*(0)$ can be determined *on-line* by solving a mixed-integer quadratic program (Bemporad and Morari, 1999), determining the solution *for all* vectors $x(0)$ within a given polytopic set $\mathcal{X}(0)$ of states of interest and *off-line* is a much harder one (Mayne, 2001; Borrelli *et al.*, 2005; Mayne and Rakovic, 2002). Once the optimal control law is obtained explicitly, on-line computation is reduced to a simple function evaluation.

The problem can be decomposed in a certain number of sub-problems that are easier to solve by exploiting the properties of the hybrid model (1). Starting from a given initial state $x(0)$ and by applying a given input sub-sequence $\{u(0),\ \ldots,\ u(k-1)\}$, the state of the system $x(k)$ belongs to a certain polyhedron $\mathcal{X}_{i(k)}$ of the partition, where $i(k)$ is the mode entered by the hybrid model at time k, $i(k) \in \{1, \ldots, s\}$. We refer to $v = \{i(0),\ \ldots,\ i(N-1)\}$ as the *switching* (or *mode*) *sequence*, and to $v^k = i(k)$ as the $(k+1)$-th element of that sequence, so that $v^k = j$

[1] The formulation and the results of this paper can be immediately extended when some of the input/state components are binary.

means that $\begin{bmatrix} x(k) \\ u(k) \end{bmatrix} \in \mathcal{X}_j$. The maximum number of possible switching sequences is $q \triangleq s^N$. Once a switching sequence v_i is fixed, system (1) is forced to enter the modes defined by v_i and becomes a linear time-varying system.

For a fixed switching sequence v_i, $i \in \{1, \ldots, q\}$, problem (2) becomes

$$J_{v_i}(x(0)) \triangleq \min_U J(U, x(0))$$
$$\text{s.t.} \quad \begin{cases} x(k+1) = A_{v_i^k} x(k) + B_{v_i^k} u(k) + f_{v_i^k}, \\ y(k) = C_{v_i^k} x(k) + D_{v_i^k} u(k) + g_{v_i^k}, \\ \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} \in \mathcal{X}_{v_i^k}, \quad k = 0, \ldots, N-1, \\ x_{\min} \le x(k) \le x_{\max} \\ y_{\min} \le y(k) \le y_{\max}, \ k = 0, \ldots, N-1, \\ u_{\min} \le u(k) \le u_{\max}, \\ x(N) \in \mathcal{X}^N. \end{cases} \tag{3}$$

Problem (3) is an optimal control problem with finite horizon N for a constrained time-varying system, and can be solved via multiparametric quadratic programming, where U are the optimization variables and $x(0) \in \mathcal{X}(0)$ are the parameters.

For all $i = 1, \ldots, q$, the solution of the optimal control problem (3) is a PPWA state feedback control law of the form (Bemporad *et al.*, 2002)

$$u_i^*(x(0)) = F_j^i x(0) + G_j^i, \quad \forall x(0) \in T_j^i, j = 1, \ldots, N_{ri} \tag{4}$$

where $\mathcal{D}_i \triangleq \bigcup_{j=1}^{N_{ri}} T_j^i$ is a convex polyhedron, corresponding to the set of states $x(0)$ for which problem (3) admits a feasible solution. The sub/superscript i in (4) means that this solution is valid for a certain fixed sequence v_i. The optimal solution $u^*(x(0))$ to Problem (2) can be found by solving problem (3) for all feasible sequences v_i, as suggested in (Mayne and Rakovic, 2002), (Mayne, 2001), and then by comparing the costs $J_{v_i}(x(0))$ on-line, given the current state $x(0)$. The optimal set $\mathcal{D}^0$ of the states $x(0)$ for which (2) admits a feasible solution is

$$\mathcal{D}_0 = \bigcup_{i=1}^{q} \mathcal{D}_i, \tag{5}$$

and, in general, is not convex.

All polyhedra T_j^i needs to be analyzed. If $T_j^i \cap T_m^l = \emptyset$ for all $l \neq i$, $l = 1, \ldots, q$, $m = 1, \ldots, N_{rl}$, then the switching sequence v_i is the only feasible one for all the states $x(0) \in T_j^i$, and so the optimal solution $u^*(x(0))$ is given by (4). We will refer to T_j^i as a *polyhedron of single feasibility*. It can happen, however, that some initial states belong to more than one set $\mathcal{D}_i$, so we need to compare the cost functions J_{v_i} in order to choose the optimal control gains (F_j^i, G_j^i). If T_j^i intersects one or more polyhedra, then the states belonging to the intersection are feasible for more than one switching sequence and the corresponding value functions need to be compared in order to compute the optimal control law. In the simple case when only two polyhedra overlap, for all states belonging to $T_j^i \cap T_m^l$ the optimal move $u^*(x(0))$ for problem (2) is

$$u^*(x(0)) = \begin{cases} F_j^i x(0) + G_j^i & \text{if} J_{v_i}^*(x(0)) < J_{v_l}^*(x(0)) \\ F_m^l x(0) + G_m^l & \text{if} J_{v_i}^*(x(0)) > J_{v_l}^*(x(0)) \\ \begin{cases} F_j^i x(0) + G_j^i \\ \text{or} \\ F_m^l x(0) + G_m^l \end{cases} & \text{if} J_{v_i}^*(x(0)) = J_{v_l}^*(x(0)). \end{cases} \tag{6}$$

A polyhedron of *multiple feasibility* [2]. on which n value functions intersect may be split into at most n possibly nonconvex subsets where in each one of them a certain value function is smaller than all the others. Because $J_{v_i}^*(x(0))(i = 1, \ldots, q)$ are quadratic functions on $T_j^i(j = 1, \ldots, N_{ri})$ the closure of the sets corresponding to the optimal state partition, in general, has the form (Borrelli *et al.*, 2005)

$$\bar{\mathcal{R}}_k^i \triangleq \{x : x' L_k^i(j) x + M_k^i(j) x \le N_k^i(j)\}. \tag{7}$$

In this paper we avoid splitting regions that overlap and storing non-polyhedral sets, but rather keep all polyhedra T_j^i for which the corresponding cost $J_{v_i}^*(x(0))$ is optimal for at least one state $x(0)$, leaving the cost comparison to the on-line procedure. This approach allows one to save memory space (no split implies less regions to store), at the price of a slightly increased on-line CPU time for the evaluation of the control move, because if $x(0)$ belongs to a region of multiple feasibility, the costs corresponding to all overlapping regions where $x(0)$ belong must be computed and compared.

4. ENUMERATION OF FEASIBLE MODE SEQUENCES VIA BACKWARDS REACHABILITY ANALYSIS

Computing the optimal solution via enumeration of all possible switching sequences can be too onerous, as the number of mp-QPs that need to be solved is $q = s^N$. Also, the set $\mathcal{D}_i$ of states $x(0)$ for which problem (3) has a solution may be empty for many switching sequences v_i.

The list of all (and only) sequences that are feasible for problem (3) can be obtained by solving a backwards reachability analysis problem as described below.

Assume that the terminal polyhedral set $\mathcal{X}^N$ is contained in one of the regions $\mathcal{X}_j$ of the polyhedral partition of system (1), i.e., $\mathcal{X}^N \subseteq \mathcal{X}_j$,

[2] In general, we say that a polyhedron T_j^i is of *multiple feasibility* if it has a non-empty intersection with one or more polyhedra $T_m^l, (i, j \neq l, m)$ belonging to a different solution of the form (4).

for some $j \in \{1, \ldots, s\}$ (in case $\mathcal{X}^N$ overlaps with more than one region of the PWA partition, one needs to consider all the nonempty intersections $\mathcal{X}_N \cap \mathcal{X}_i$, $i = 1, \ldots, s$). Next, for each mode $i = 1, \ldots, s$ we determine which polyhedral subsets of $\mathbb{R}^{n+m}$ defined by the linear inequalities

$$\begin{cases} A_i x_{N-1} + B_i u_{N-1} + f_i \in \mathcal{X}^N \\ \left[x'_{N-1}\ u'_{N-1}\right]' \in \mathcal{X}_i \\ x_{\min} \leq x_{N-1} \leq x_{\max} \\ y_{\min} \leq C_i x_{N-1} + D_i u_{N-1} + g_i \leq y_{\max} \\ u_{\min} \leq u_{N-1} \leq u_{\max} \end{cases} \tag{8}$$

are nonempty (this just requires a phase-1 of a linear program). Let $\mathcal{X}_j^{N-1}$, $j = 1, \ldots, k_{N-1}$ be such nonempty sets. At the next step of the backwards reachability analysis, for each $j = 1, \ldots, k_{N-1}$ and for each mode $i = 1, \ldots, s$ we determine which polyhedral subsets of $\mathbb{R}^{n+2m}$ defined by the linear inequalities

$$\begin{cases} \left[(A_i x_{N-2} + B_i u_{N-2} + f_i)'\ u'_{N-1}\right]' \in \mathcal{X}_j^{N-1} \\ \left[x'_{N-2}\ u'_{N-2}\right]' \in \mathcal{X}_i \\ x_{\min} \leq x_{N-2} \leq x_{\max} \\ y_{\min} \leq C_i x_{N-2} + D_i u_{N-2} + g_i \leq y_{\max} \\ u_{\min} \leq u_{N-2} \leq u_{\max} \end{cases} \tag{9}$$

are nonempty. By letting $\mathcal{X}_j^{N-2}$, $j = 1, \ldots, k_{N-2}$ be such nonempty sets, the procedure is repeated backwards until the time index reaches 0.

The switching sequences $v_1, \ldots, v_{\bar{q}}$, where $\bar{q} = k_0 \leq s^N$ are all and only the switching sequences for system (1) that satisfy the constraints in (3) for at least one initial state $x(0)$ and input sequence $u(0), \ldots, u(N-1)$, and that will be referred to as the *feasible* switching sequences.

The above procedure is successfully implemented in the Hybrid Toolbox for Matlab (Bemporad, 2003).

5. COST COMPARISONS AND REGION ELIMINATION

The main problem (FTCOC) has been decomposed in $\bar{q}$ subproblems, depending on the number of feasible switching sequences.

Every subproblem (3), once solved via multiparametric quadratic programming, gives a PPWA control law of the form (4) and an associated optimal cost function $J^*_{v_i}(x(0))$ that is convex, continuous, and piecewise quadratic (PWQ) on the same partition.

By solving the problem for every feasible switching sequence v_i, we obtain $\bar{q}$ state partitions $\mathcal{D}_i$ that need to be compared in order to find the optimal solution of Problem (2). For a given $x(0)$, the optimal input $u^*(0)$ is obtained comparing every cost function $J^*_{v_1}(x(0)), , \ldots, J^*_{v_{\bar{q}}}(x(0))$, and find the associated control input at minimum cost. The main problem is that, in the worst case, the number of possible comparisons that need to be made on line in order to find the minimum cost is $(\bar{q} - 1)$, and so the main advantage of saving on-line CPU time by calculating the control law off line may be lost. In addition, typically there are several regions whose associated control law is never the optimal one. A region of multiple feasibility T_j^i is *dominated* if

$$\begin{array}{l} \forall x \in T_j^i, \exists l \in \{1, \ldots, \bar{q}\}, m \in \{1, \ldots, N_{rl}\} : \\ \quad x \in T_j^i \cap T_m^l, J^*_{v_l}(x) < J^*_{v_i}(x). \end{array} \tag{10}$$

otherwise it is considered *optimal*, since it exists at least one vector where its corresponding function $J^*_{v_i}(x)$ is optimal. It is desirable to eliminate all dominated regions T_j^i and the related cost functions in order to avoid a useless waste of CPU time for searching the region with minimum cost, and of memory for storing dominated regions. In other words, we want to keep only the regions T_j^i that are certainly optimal in a certain subset of the state set $\mathcal{D}(0)$.

5.1 Determination of Polyhedra of Single Feasibility

We first locate all the regions T_j^i for which

$$\exists \bar{x} \in \mathbb{R}^m : \bar{x} \in T_j^i, \bar{x} \notin T_m^l, \forall l \neq i,\ l = 1, \ldots, \bar{q}, \tag{11}$$

where clearly $\bar{x} \in T_j^i \Rightarrow \bar{q} \notin T_m^i$, $\forall m = 1, \ldots, N_{ri}$, $m \neq j$.

Regions T_j^i satisfying (11) do not need to be tested for domination by other regions, as they are clearly optimal in at least one point $\bar{x} \in \mathbb{R}^m$. Condition (11) can be tested by solving the following MILP for all $i = 1, \ldots, \bar{q}$, $\forall j = 1, \ldots, N_{ri}$:

$$\begin{array}{ll} \min\limits_{x,\delta} & 0 \\ \text{s.t.} & \sum\limits_{r=1}^{N_h} \delta_{hr} \leq N_h - 1, \quad \forall h = 1, \ldots, \bar{q},\ h \neq i \\ & A_j^i x \leq b_j^i, \\ & H_h^r x - K_h^r > m\delta, \\ & H_h^r x - K_h^r \leq M(1-\delta), \\ & \delta_{hr} \in \{0, 1\},\ \forall h = 1, \ldots, \bar{q},\ \forall r = 1, \ldots, N_h \end{array} \tag{12}$$

where $\bar{q}$ is the total number of *envelopes* $\mathcal{D}_i$ (that is, of the switching sequences v_i for which (3) is feasible), (A_j^i, b_j^i) defines the region T_j^i that needs to be tested, N_h represents the total number of facets of the envelope of the h-th partition $\mathcal{D}_h$, and $r = 1, \ldots, N_h$ is its r-th facet and finally m, M are chosen such that

$$m < \min_{x \in \mathcal{X}(0)} H_h^r x - K_h^r, \quad M \geq \max_{x \in \mathcal{X}(0)} H_h^r x - K_h^r.$$

The binary variables δ_{hr} satisfy the condition $[\delta_{hr} = 1] \leftrightarrow [H_h^r x \leq K_h]$, i.e., $\delta_{hr} = 1$ iff $x \in T_j^i$

satisfies the constraint that defines that r-th facet of the h-th envelope $\mathcal{D}_h$, and the first constraint in (12) imposed that at least one facet inequality is violated, so that $x \notin \mathcal{D}_h$. Regions T^i_j for which (12) is feasible are regions of single feasibility and therefore optimal, so they must be retained in the final hybrid MPC control law.

5.2 DC Programming Approach

In order to find the optimal regions, we need to compare quadratic functions over certain convex sets of parameters, for this reason it can be recast as a DC (Difference of Convex functions) problem (Horst and Thoai, 1999).

For every region T^l_m for which (12) is infeasible, we need to determine all the partitions $\mathcal{D}_k$, $k \neq l$, such that $T^l_m \cap \mathcal{D}_k \neq \emptyset$. Clearly, $T^l_m \subseteq \bigcup_{k \neq l} \mathcal{D}_k$. In general, T^l_m intersects $s^l_m < \bar{q}$ partitions $\mathcal{D}_k$. By letting $\mathcal{S} = \{1, .., \bar{q}\}$, we will refer to $\mathcal{S}^l_m$ as the subset of $\mathcal{S}$ of indices $k \neq l$, such that $T^l_m \cap \mathcal{D}_k \neq \emptyset$. Clearly $s^l_m = \text{card}\ (\mathcal{S}^l_m)$. We assume here for simplicity that $T^l_m \subseteq \mathcal{D}_k$, $\forall k \in \mathcal{S}^l_m$. This assumption will be removed shortly.

For every fixed switching sequence v_l, the optimal solution $J^*_{v_l}$ obtained by solving the associated mpQP problem (3) is piecewise quadratic (PWQ) over the polyhedral partition $\mathcal{D}_l$, and quadratic in every single region T^l_m. In the sequel we will refer to $V^l_m(x)$ as the quadratic term of the value function $J^*_{v_l}$ in the m-th region of the l-th partition. A region T^l_m is *optimal* if

$$\exists x^* \in T^l_m : V^l_m(x^*) - J^*_{v_k}(x^*) < 0, \forall k \in \mathcal{S}^l_m. \quad (13)$$

Condition (13) can be verified by solving the following DC programming problem for all $k \in \mathcal{S}^l_m$

$$T^*_{lmk} = \min_{x \in T^l_m} V^l_m(x) - J^*_{v_k}(x). \quad (14)$$

If $T^*_{lmk} > 0$ for some $k \in \mathcal{S}^l_m$, region T^i_j is certainly dominated (i.e., not optimal) and can be safely discarded. The DC problem (14) is a nonconvex problem. On the other hand, we do not necessarily need to find its optimal solution, but a positive lower bound on the minimum would suffice for checking condition (13). In the next sections we describe a procedure for computing such a lower bound in an arbitrarily tight manner.

In the more general case where $T^l_m \nsubseteq \mathcal{D}_k$ for some $k \in \mathcal{S}^l_m$, for all such indices k the quadratic and piecewise quadratic costs are compared over the subset $\Omega^{l,k}_m \triangleq T^l_m \cap \mathcal{D}_k$. In this case, one can conclude that the region T^l_m is dominated if and only if all its subsets $\Omega^{l,k}_m$ are dominated.

5.3 DC Algorithm

In order to simplify the notation, given a region T^l_m of multiple feasibility and a partition $\mathcal{D}_k$, $k \in \mathcal{S}^l_m$, we will refer to $V^l_m(x)$ and $J^*_{v_k}(x)$ as $f_1(x)$ and $f_2(x)$, respectively.

Now, suppose to compute two PPWA functions $\bar{f}_1$ and $\bar{f}_2$ such that

$$\exists \epsilon_i > 0 : f_i(x) \leq \bar{f}_i(x),\ \epsilon_i = \max_{x \in \Omega^{l,k}_m} (f_i(x) - \bar{f}_i(x)),\ i = 1, 2. \quad (15)$$

Clearly, the following relations

$$f_1(x) - \bar{f}_2(x) \leq f_1(x) - f_2(x) \leq \bar{f}_1(x) - f_2(x) \quad (16)$$

are verified $\forall x \in \Omega^{l,k}_m$.

Now define $LB_k, UB_k \in \mathbb{R}$ as the solutions of the quadratic programs

$$LB_k = \min_{x \in \Omega^{l,k}_m} J_1(x) = f_1(x) - \bar{f}_2(x) \quad (17a)$$

$$UB_k = \max_{x \in \Omega^{l,k}_m} J_2(x) = \bar{f}_1(x) - f_2(x) \quad (17b)$$

By (16)–(17), it follows that

$$LB_k \leq \min_{x \in \Omega^{l,k}_m} f_1(x) - f_2(x) \leq UB_k, \quad \forall k \in \mathcal{S}^l_m. \quad (18)$$

If $UB_k < 0$, $\forall k \in \mathcal{S}^l_m$, then Condition (13) is satisfied (region T^l_m is optimal), while if $LB_{\bar{k}} > 0$ for some $k \in \mathcal{S}^l_m$, then T^l_m is dominated by partition $\mathcal{D}_k$. In the other cases, one needs to subpartition $\Omega^{l,k}_m$ in order to obtain tighter upper and lower bounds, as described in the following algorithm.

---

function SignTest ($f1$, $f2$, $\Omega^{l,k}_m$)

(1) Obtain an initial triangulation of $\Omega^{l,k}_m$ in simplices S_i, ($i = 1, \ldots, N_s$) via Delaunay triangulation (Yepremyan and Falk, 2005);
(2) Optimal := True;
(3) For ($k \in \mathcal{S}^l_m$) and (Optimal=True) Do,
  (a) Optimal := False;
  (b) For $i = 1$ to N_s Do, *loops over S_i *\
    (i) $\{LB_k, UB_k\}$ = Bounds(S_i, True);
      * Compute UB_k, LB_k as in (17a)-(17b) over S_i; *\
    (ii) If $UB_k < 0$ then
      Optimal=true; * $f_1(x) < f_2(x), \forall x \in S_i$ *\
      Choose another partition $\mathcal{D}_k$, $k \in \mathcal{S}^l_m$;
      break;
(4) If (Optimal = False) then 'T^l_m is dominated by $\mathcal{D}_k$, otherwise T^l_m is optimal

---

function Bounds(Simplex S,boolean turn)

(1) If (Turn = true)
  (a) Solve problem (17a).
  (b) Set LB = LB_k and $\bar{x} = x^* = \arg\min_x J_1(x)$.
  (c) If $LB > 0$, return;
  (d) for (k=0 to n) do
    (i) Substitute the k-th vertex of S with $\bar{x}$, and obtain $n+1$ new simplices $S_0, \ldots, S_n$;

```
      (ii) Turn := false;
      (iii) Bounds(S_k, Turn);

(2) Otherwise
   (a) Solve problem (17b).
   (b) Set UB = UB_k and x̄ = x* = arg max_x J_2(x).
   (c) If UB < 0, stop
       the region is optimal;
   (d) Else for (k=0 to n) do
       (i) Substitute the k-th vertex of S with x̄,
           and obtain n+m+1 new simplices S_0,...,S_n
       (ii) Turn := true;
      (iii) Bounds(S_k, Turn);
```

Remark 1. The algorithm computes an initial simplicial partition $S_0, .., S_n$ of the given set $\Omega_m^{l,k}$ and solves the two QPs defined in (17a)-(17b) over $S_i,\ i = 0, .., n$. Whatever none of the two conditions is satisfied over the current simplex, it proceeds recursively, by splitting every simplex into $n + 1$ simplices, adding a new vertex $\bar{x} = \arg\min_x\{J_1(x), \max_x J_2(x)\}$, until any of the conditions (17a)-(17b) is satisfied. The algorithm stops splitting the initial set of simplices when it finds a point where $UB < 0$. Note that the generated simplices are only needed to compare cost functions, and hence are discarded immediately after the comparison. In particular, they are not at all needed to store the control law.

5.4 Upper-approximation of the Value Function

Under the assumptions made in (3), the optimal k-th mpQP solution $J^*_{v_k}(x)$, $k \in \mathcal{S}_m^l$, is a convex (piecewise quadratic) function, defined over the convex full-dimensional set of parameters $\mathcal{D}_k \subseteq \mathbb{R}^n$ (Mangasarian and Rosen, 1964). A complete reference for the algorithm used for computing an upper-approximation in piecewise affine form over a simplicial partition of a convex (piecewise quadratic) function can be found in (Bemporad and Filippi, 2006)

5.5 Reduction of Partially Dominated Regions

When a region is only *partially* dominated, that is, if a polyhedral subset $\bar{\Omega}$ of a *optimal* region T_m^l is dominated by a certain partition $\mathcal{D}_k$, it may be desirable to reduce T_m^l to a smaller region that does not contain $\bar{\Omega}$.

Definition 1. A matrix-vector pair (A, b) is a *minimal representation* of a polyhedron $P = \{x : Ax \leq b\}$, if there does not exist a pair (A_1, b_1) defining the same polyhedron and such that $\dim(b_1) < \dim(b)$.

Lemma 1. Given two nonempty polyhedra P, Q and their minimal representations $P = \{x \in \mathbb{R}^n : Ax \leq b\}, Q = \{x \in \mathbb{R}^n : Cx \leq d\}$, the set $P \backslash Q \triangleq \{x \in \mathbb{R}^n : x \in P, x \notin Q\}$ is nonconvex if and only if the number of hyperplanes $c_j x \leq d_j \in \partial G$ which intersect the interior of P is greater or equal than two.

Proof: Suppose that $c_1' x = d_1$ and $c_2' x = d_2$ are two hyperplanes of Q which intersect the interior of P in two points x_1, x_2, respectively.

Let $[x_1, x_2]$ denote the line segment $\{x \in \mathbb{R}^n : x = \lambda x_1 + (1-\lambda)x_2, 0 \leq \lambda \leq 1\}$, which is entirely contained in Q. Since x_1, x_2 belong to the interior of P then there exist two scalars $\theta_1 > 1$, $\theta_2 < 0$ such that $\bar{x}_1$ and $\bar{x}_2$ defined as $\bar{x}_1 = \theta_1 x_1 + (1-\theta_1)x_2, \bar{x}_2 = \theta_2 x_1 + (1-\theta_2)x_2$ belong to the interior of P. Since $c_1'\bar{x}_1 = \theta_1 c_1' x_1 + (1-\theta_1)c_1' x_2 > \theta_1 d_1 + (1-\theta_1)d_1 = d_1$, then $\bar{x}_1 \notin Q$, and similarly one can show that $x_2 \notin Q$. Hence, $\bar{x}_1,\ \bar{x}_2 \in P\backslash Q$. Consider the convex combination of $\bar{x}_1, \bar{x}_2$ defined as $\gamma\bar{x}_1 + (1-\gamma)\bar{x}_2$, $\gamma \in (0, 1)$. We want to show that there exists a $0 < \gamma < 1$ such that $\gamma\bar{x}_1 + (1-\gamma)\bar{x}_2 \in Q$, and hence does not belong to $P\backslash Q$. Since $\gamma\bar{x}_1 + (1-\gamma)\bar{x}_2 = \gamma(\theta_1 x_1 + (1-\theta_1)x_2) + (1-\gamma)(\theta_2 x_1 + (1-\theta_2)x_2) = (\gamma\theta_1 + (1-\gamma)\theta_2)x_1 + (1-(\gamma\theta_1 + (1-\gamma)\theta_2))x_2$ is a linear combination of x_1 and x_2, and since the open segment (x_1, x_2) is contained in the interior of Q, there exists $\bar{\alpha}$ such that $x = \bar{\alpha}x_1 + (1-\bar{\alpha})x_2 \in Q,\ 0 < \bar{\alpha} < 1\}$. By setting $\bar{\alpha} = \gamma\theta_1 + (1-\gamma)\theta_2$ and by choosing any γ such that

$$0 < \frac{-\theta_2}{\theta_1 - \theta_2} < \gamma < \frac{1-\theta_2}{\theta_1 - \theta_2} < 1 \qquad (19)$$

it follows that $x \notin P\backslash Q$, which proves that $P\backslash Q$ is not convex.

In the same way we can show that $P\backslash Q$ is not convex if the number of hyperplanes is more than two, since it is enough to repeat the above argument for every pair of inequalities defined by $(c_i, d_i), (c_k, d_k)$. On the other hand, if only one hyperplane of Q intersects the interior of P the resulting set $P\backslash Q$ is the intersection of convex sets, and therefore convex, or if no hyperplane of Q intersects P then $P\backslash Q = P$ is also convex. □

Thanks to Lemma 1, one can reduce all regions T_m^l that are partially dominated by partitions $\mathcal{D}_k$ that intersect T_m^l with at most two hyperplanes. In this way, on-line computations are possibly simplified because of the reduced overlaps among the regions of the controller's partition.

6. EXAMPLE

Consider the following system

$$x(k+1) = \begin{cases} A_1 x(k) + B_1 u(k) & \text{if } x_2(k) + x_3(k) < 0, \\ & \quad |x_1(k)| < 2 \\ A_2 x(k) + B_2 u(k) & \text{if } x_2(k) + x_3(k) \geq 0, \\ & \quad |x_1(k)| < 2 \\ A_3 x(k) + B_3 u(k) & \text{otherwise} \end{cases} \tag{20}$$

where $x(k) \in \mathbb{X} = [-10, 10]^3$, $u(k) \in \mathbb{U} = [-2, 2]$, $A_1 = \left[\begin{smallmatrix} 1 & .4 & .08 \\ 0 & 1 & .4 \\ 0 & 0 & 1 \end{smallmatrix}\right]$, $A_2 = \left[\begin{smallmatrix} 1 & .7 & .245 \\ 0 & 1 & .7 \\ 0 & 0 & 1 \end{smallmatrix}\right]$, $A_3 = \left[\begin{smallmatrix} 1 & .8 & .32 \\ 0 & 1 & .8 \\ 0 & 0 & 1 \end{smallmatrix}\right]$, $B_1 = [\,.0107 \ .08 \ .4\,]$, $B_2 = [\,.0572 \ .245 \ .7\,]$, and $B_3 = [\,.0853 \ .32 \ .8\,]$. The PWA system has three dynamic modes, defined over 6 regions.

We want to regulate the state of the system to the origin, and find the explicit control law using the quadratic cost defined by the weights $Q = I$, $P = I$, $R = .1$, and control horizon $N = 3$. We obtain 119 feasible switching sequences, instead of the $6^3 = 216$ possible ones, and 632 polyhedral regions T^l_m. The preliminary inclusion test (12) finds 129 regions of single feasibility. The remaining 503 regions T^l_m of multiple feasibility need to be compared with the corresponding s^l_m partitions $\mathcal{D}_k$ in order to detect their optimality. After running the algorithm described in Section 5.3, 283 regions are found to be totally dominated while 36 can be reduced by using the results of Lemma 1. In this way we have reduced the number of regions in the final control law by 40%, therefore decreasing the number of comparisons that needs to be made on line, without any loss of optimality.

7. CONCLUSIONS

In this paper we have proposed an approach for solving hybrid optimal control problems based on quadratic costs explicitly with respect to the initial state. The method lists all feasible switching sequences using backwards reachability analysis, solves the associated multiparametric quadratic programs, and then reduces the total number of regions via a comparison of the value functions. The latter is computed by using a recursive partition of the parameter space in simplices, by making a linear approximation of the convex value functions in each simplex, and by calculating an upper and a lower bound to their difference. The procedure allows one to discard all those regions whose associated value function is never optimal.

REFERENCES

Baotic, M., F.J. Christophersen and M. Morari (2003). Infinite time optimal control of hybrid systems with a linear performance index. In: *Proc. 42th IEEE Conf. on Decision and Control*. Maui, Hawaii, USA. pp. 3191–3196.

Bemporad, A. (2003). *Hybrid Toolbox – User's Guide*. `http://www.dii.unisi.it/hybrid/toolbox`.

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

Bemporad, A. and C. Filippi (2006). Approximate multiparametric convex programming. *Computational Optimization and Applications*. to appear.

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

Bemporad, A., F. Borrelli and M. Morari (2000). Piecewise linear optimal controllers for hybrid systems. in American Control Conference. Chicago, IL. pp. 1190–1194.

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

Borrelli, F., M. Baotic, A. Bemporad and M. Morari (2005). Dynamic programming for constrained optimal control of discrete-time linear hybrid systems. *Automatica* **41**(10), 1709–1721.

Corona, D. (2005). Optimal Control of Linear Affine Hybrid Automata. PhD thesis. Dipartimento di Ingegneria Elettrica ed Elettronica. University of Cagliari, Italy.

Horst, R. and N.V. Thoai (1999). DC programming: Overview. *Journal of Optimization Theory and Applications* **103**(1), 1–43.

Mangasarian, O.L. and J.B. Rosen (1964). Inequalities for stochastic nonlinear programming problems. *Operations Research* **12**, 143–154.

Mayne, D.Q. (2001). Constrained optimal control. European Control Conference, Plenary Lecture.

Mayne, D.Q. and S. Rakovic (2002). Optimal control of constrained piecewise affine discrete time systems using reverse transformation. *Conference on Decision and Control, Las Vegas, Nevada* pp. 1564–1551.

Yepremyan, L. and J.E. Falk (2005). Delaunay partitions in $\mathbb{R}^n$ applied to non-convex programs and vertex/facet enumeration problems. *Computers and Operations Research* (32), 793–812.