JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS: Vol. 118, No. 3, pp. 515–540, September 2003 (© 2003)

Geometric Algorithm for Multiparametric Linear Programming[1]

F. Borrelli,[2] A. Bemporad,[3] and M. Morari[4]

Communicated by O. L. Mangasarian

Abstract. We propose a novel algorithm for solving multiparametric linear programming problems. Rather than visiting different bases of the associated LP tableau, we follow a geometric approach based on the direct exploration of the parameter space. The resulting algorithm has computational advantages, namely the simplicity of its implementation in a recursive form and an efficient handling of primal and dual degeneracy. Illustrative examples describe the approach throughout the paper. The algorithm is used to solve finite-time constrained optimal control problems for discrete-time linear dynamical systems.

Key Words. Multiparametric programming, sensitivity analysis, post-optimality analysis, linear programming, optimal control.

1. Introduction

The operations research community has addressed parameter variations in mathematical programs at two levels: sensitivity analysis, which characterizes the change of the solution with respect to small perturbations of the parameters; and parametric programming, where the characterization of the solution for a full range of parameter values is sought. More precisely, programs which depend on only one scalar parameter are referred to as parametric programs, while problems depending on a vector of parameters are referred to as multiparametric programs.

[1]The authors gratefully thank Dr. Carlo Filippi for his invaluable comments.

[2]Research Associate, Automatic Control Laboratory, Swiss Federal Institute of Technology, ETH, Zürich, Switzerland.

[3]Assistant Professor, Dipartimento di Ingegneria dell'Informazione, Università di Siena, Siena, Italy.

[4]Professor, Automatic Control Laboratory, Swiss Federal Institute of Technology, ETH, Zürich, Switzerland.

The first method for solving parametric linear programs was proposed by Gass and Saaty (Ref. 1); since then, extensive research has been devoted to sensitivity and (multi)parametric analysis, as testified by the hundreds of references in Ref. 2; see also Ref. 3 for recent advances in the field.

The first method for solving multiparametric linear programs was formulated by Gal and Nedoma (Ref. 4). Subsequently, only a few authors have dealt with multiparametric linear (Refs. 2, 5, 6), nonlinear (Ref. 7), quadratic (Ref. 8), and mixed-integer linear (Ref. 9) programming solvers.

There are several reasons to look for efficient solvers of multiparametric programs. Typically, mathematical programs are affected by uncertainties due to factors which are either unknown or that will be decided only later. Parametric programming subdivides systematically the space of parameters into characteristic regions, which depict the feasibility and corresponding performance as a function of the uncertain parameters and hence provide the decision maker with a complete map of various outcomes.

Our interest in multiparametric programming arises from the field of system theory and optimal control. Optimal control problems for constrained discrete-time systems based on linear programming were formulated in the early 1960s by Zadeh and Whalen (Ref. 10), but a procedure to solve such problems in an explicit state-feedback form was never proposed. By using multiparametric linear programming, we can compute efficiently the feedback control law for such an optimal control problem as shown in Section 4.

Moreover, our interest arises from the so-called model predictive control (MPC) technique. MPC is very popular in the process industries for the automatic regulation of process units under operating constraints (Ref. 11) and has attracted a considerable research effort in the last decade, as surveyed recently in Ref. 12. MPC requires an optimization problem to be solved online in order to compute the next command action. Such an optimization problem depends on the current sensor measurements. The computation effort can be moved offline by solving multiparametric programs, where the command inputs are the optimization variables and the measurements are the parameters (Refs. 13–14).

In this article, we focus on multiparametric linear programming. We propose a new algorithm which, rather than visiting the different bases of the associated LP tableau (Ref. 4), is based on the direct exploration of the parameter space (Ref. 8).

Multiparametric analysis depends on the concept of critical region. In Ref. 4, a critical region is defined as a subset of the parameter space on which a certain basis of the linear program is optimal. The algorithm proposed in Ref. 4 for solving multiparametric linear programs generates non-overlapping critical regions by generating and exploring the graph of bases.

In the graph of bases, the nodes represent optimal bases of the given multiparametric problem and two nodes are connected by an edge if it is possible to pass from one basis to another by one pivot step (in this case, the bases are called neighbors).

Our definition of critical regions is not associated with the bases but with the set of active constraints and is related directly to the definition given in Refs. 3, 5, 15, 16. In the absence of degeneracy, our algorithm explores implicitly the graph of bases without resorting to pivoting. In case of degeneracy, it avoids visiting the graph of degenerate bases. Therefore, the approach is different from other methods based on the simplex tableau (Ref. 2).

The resulting algorithm for solving multiparametric linear programs has computational advantages, namely the simplicity of its implementation in a recursive form and the possibility to look for parametric solutions within a given polyhedral region of the parameter space without solving the problem globally. The algorithm is capable of handling primal and dual degeneracy. Polynomial time LP solvers can be used during its execution.

2. Multiparametric Linear Programming

Consider the right-hand side multiparametric linear program (MPLP)

$$\min_{x} \quad z = c^T x, \tag{1a}$$

$$\text{s.t.} \quad Gx \leq w + F\theta, \tag{1b}$$

where $x \in \mathbb{R}^n$ is the optimization vector, $\theta \in \mathbb{R}^s$ is the vector of parameters, $z \in \mathbb{R}$ is the objective function, and $G \in \mathbb{R}^{m \times n}$, $c \in \mathbb{R}^n$, $w \in \mathbb{R}^m$, and $F \in \mathbb{R}^{m \times s}$. The reason for choosing the linear program structure (1) is motivated by its application to optimal control problems, as discussed in Section 4.

Given a polyhedral set $K \subseteq \mathbb{R}^s$ of parameters,

$$K \triangleq \{\theta \in \mathbb{R}^s | S\theta \leq q\}, \tag{2}$$

we denote by $K^* \subseteq K$ the region of parameters $\theta \in K$ such that the LP (1) has a finite optimal solution.

For any given $\bar{\theta} \in K^*$, let $z^*(\bar{\theta})$ denote the minimum value of the objective function in problem (1) for $\theta = \bar{\theta}$. The symbol $z^*: K^* \to \mathbb{R}$ will denote the function expressing the dependence on θ of the minimum value of the objective function over K^* and will be called the value function. The set-valued function $X^*: K^* \to 2^{\mathbb{R}^n}$, where $2^{\mathbb{R}^n}$ is the set of subsets of $\mathbb{R}^n$, will describe the set of optimizers $x^*(\theta)$ related to $z^*(\theta)$ for any fixed $\theta \in K^*$. We give the following definition of primal and dual degeneracy.

Definition 2.1. For any given $\theta \in K^*$, the LP (1) is said to be primal degenerate if there exists an $x^*(\theta) \in X^*(\theta)$ such that the number of active constraints at the optimizer is greater than the number of variables n.

Definition 2.2. For any given $\theta \in K^*$, the LP (1) is said to be dual degenerate if its dual problem is primal degenerate.

We aim at determining the feasible region K^* of parameters, the expression of the value function, and the expression of one of the optimizers $x^*(\theta) \in X^*(\theta)$.

Rather than solving the problem by exploring the graph of bases of the associated LP tableau (Refs. 2, 4), our approach is based on the direct exploration of the parameter space (Ref. 8). As will be detailed in the following sections, this has the following advantages: (i) degeneracy can be handled in a simpler way; (ii) the algorithm is easily implementable recursively; and (iii) the main ideas of the algorithm generalize to nonlinear multiparametric programming (Refs. 8, 9).

Multiparametric analysis depends on the extension of the concept of critical region (CR). In Ref. 4, a critical region is defined as a subset of all the parameters θ for which a certain basis is optimal for problem (1). Below, we give a definition of optimal partition and critical region related directly to that of Filippi (Ref. 5), which is the extension of the idea in Refs. 15, 16 to the multiparametric case.

Let $J \triangleq \{1, \ldots, m\}$ be the set of constraint indices. For any $A \subseteq J$, let G_A and F_A be the submatrices of G and F, respectively, consisting of the rows indexed by A, and let G_j, F_j, w_j be the jth rows of G, F, w.

Definition 2.3. The optimal partition of J associated with $\theta \in K^*$ is the partition $(A(\theta), NA(\theta))$ where

$$A(\theta) \triangleq \{j \in J \mid G_j x^*(\theta) - F_j \theta = w_j, \text{ for all } x^*(\theta) \in X^*(\theta)\},$$

$$NA(\theta) \triangleq \{j \in J \mid G_j x^*(\theta) - F_j \theta < w_j, \text{ for some } x^*(\theta) \in X^*(\theta)\}.$$

It is clear that $(A(\theta), NA(\theta))$ are disjoint and that their union is J. For a given $\theta^* \in K^*$, let $(A, NA) \triangleq (A(\theta^*), NA(\theta^*))$ and let

$$\mathrm{CR}_A \triangleq \{\theta \in K \mid A(\theta) = A\}, \tag{3a}$$

$$\overline{\mathrm{CR}}_A \triangleq \{\theta \in K \mid A(\theta) \subseteq A\}. \tag{3b}$$

The set CR_A is the critical region related to the set of active constraints A, i.e., the set of all parameters θ such that the constraints indexed by A are active at the optimum of problem (1). Clearly, $\overline{\mathrm{CR}}_A \supseteq \mathrm{CR}_A$.

The following result was proved by Filippi (Ref. 5).

Theorem 2.1. Let $(A, NA) \triangleq (A(\theta^*), NA(\theta^*))$, for some $\theta^* \in K$, and let d be the dimension of range $G_A \cap$range F_A. If $d = 0$, then $\mathrm{CR}_A = \{\theta^*\}$. If $d > 0$, then:

(i) CR_A is an open[5] polyhedron of dimension d;[6]
(ii) $\overline{\mathrm{CR}}_A$ is the closure of CR_A;
(iii) every face of $\overline{\mathrm{CR}}_A$ takes the form of $\overline{\mathrm{CR}}_{A'}$ for some $A' \subseteq A$.

By Theorem 2.1 and the definition of critical regions in (3), it follows that the set K^* is always partitioned in a unique way. On the contrary, in the case of degeneracies, in the approach of Ref. 2 the partition is not defined uniquely, as it can be generated in a possibly exponential number of ways, depending on the particular path followed by the algorithm to visit different bases (Refs. 15, 17).

In this paper, we aim at determining all the full-dimensional critical regions contained in K^*, defined as in (3).

Before going further, we recall some well-known properties of the value function $z^*(\theta)$: $\mathbb{R}^s \to \mathbb{R}$ and the set K^*.

Theorem 2.2. See Ref. 2, p. 178, Theorem 1. Assume that, for a fixed $\theta^0 \in K$, there exist a finite optimal solution $x^*(\theta_0)$ of (1). Then, for all $\theta \in K$, problem (1) has either a finite optimum or no feasible solution.

Theorem 2.3. See Ref. 2, p. 179, Theorem 2. Let $K^* \subseteq K$ be the set of all parameters θ such that the LP (1) has a finite optimal solution $x^*(\theta)$. Then, K^* is a closed polyhedral set in $\mathbb{R}^s$.

We recall also the following definitions.

Definition 2.4. A collection of sets $R_1, \ldots, R_N$ is a partition of a set Θ if (i) $\bigcup_{i=1}^N R_i = \Theta$, (ii) $R_i \cap R_j = \emptyset$, $\forall i \neq j$. Moreover $R_1, \ldots, R_N$ is a polyhedral partition of a polyhedral set Θ if $R_1, \ldots, R_N$ is a partition of Θ and the closure $\bar{R}_i$ of R_i is a polyhedron for all $i = 1, \ldots, N$.

Definition 2.5. A function $h(\theta)$: $\Theta \to \mathbb{R}^k$, where $\Theta \subseteq \mathbb{R}^s$ is a polyhedral set, is piecewise affine (PWA) if there exists a polyhedral partition

[5]Given the polyhedron $B\xi \leq v$, we call open polyhedron the set $B\xi < v$.
[6]The dimension of a polyhedron $\mathscr{P}$ is defined here as the dimension of the smallest affine subspace containing $\mathscr{P}$.

$R_1, \ldots, R_N$ of Θ and $h(\theta) = H^i\theta + k^i$, $\forall \theta \in R_i$, $i = 1, \ldots, N$, where $H^i \in \mathbb{R}^{k \times s}$, $k^i \in \mathbb{R}^k$.

The following theorem summarizes the properties enjoyed by the multiparametric solution; see Ref. 2, p. 180.

Theorem 2.4. The function $z^*(\cdot)$ is convex and piecewise affine over K^* (in particular, affine in each critical region).

If the optimizer $x^*(\theta)$ is unique for all $\theta \in K^*$, then the optimizer function $x^*\colon K^* \to \mathbb{R}^n$ is continuous and piecewise affine. Otherwise, it is possible always to define a continuous and piecewise affine optimizer function $x^*(\theta) \in X^*(\theta)$ for all $\theta \in K^*$.

In the next section, we describe an algorithm to determine the set K^*, its partition into full-dimensional critical regions CR_{A_i}, the PWA value function $z^*(\cdot)$, and a PWA optimizer function $x^*(\cdot)$.

3. Geometric Algorithm for MPLP

The algorithm consists of two main steps, which can be summarized as follows:

Step 1. Determine the dimension $s' \leq s$ of the smallest affine subspace $\mathscr{K}$ that contains K^*. If $s' < s$, find the linear equalities in θ which define $\mathscr{K}$.

Step 2. Determine the partition of K^* into full-dimensional critical regions CR_{A_i}; find the function $z^*(\cdot)$ and a piecewise affine optimizer function $x^*(\cdot)$.

Below, we give the details of the two steps. The first step is preliminary and its goal is to reduce the number of parameters in order to obtain a full-dimensional feasible region of parameters. This eases the second step, which computes the multiparametric solution and represents the core of the MPLP algorithm.

3.1. Determining the Affine Subspace $\mathscr{K}$. In order to work with a minimal dimension of the parameter vector, the first step of the algorithm aims at finding the affine subspace $\mathscr{K} \subseteq \mathbb{R}^s$ containing the parameters θ which render (1) feasible.

A first simple but important consideration concerns the column rank r_F of F. Clearly, if $r_F < s$, $s - r_F$ parameters can be eliminated by a simple coordinate transformation in $\mathbb{R}^s$. Therefore, from now on, without loss of generality we will assume that F has full column rank.

Even if the matrix F has full column rank, the polyhedron K^* can be contained in a subspace of dimension $s' < s$. Therefore, before solving the MPLP problem, we need a test for checking the minimal dimension s' of the affine subspace $\mathscr{K}$ that contains K^*. Moreover, when $s' < s$, we need the equations describing $\mathscr{K}$ in $\mathbb{R}^s$. The equations are then used for a change of coordinates in order to reduce the number of parameters from s to s'.

Recall (1) and construct the LP problem in the space $\mathbb{R}^{n+s}$,

$$\min_x \quad z = c^T x, \tag{4a}$$

$$\text{s.t.} \quad Gx - F\theta \leq w. \tag{4b}$$

Clearly, the constraints in (4) define a polyhedron $\mathscr{P}$ in $\mathbb{R}^{n+s}$. The following proposition shows that the projection $\Pi_{\mathbb{R}^s}(\mathscr{P})$ of $\mathscr{P}$ on the parameter space $\mathbb{R}^s$ is K^*.

Proposition 3.1.

$$\theta^* \in K^* \Leftrightarrow \exists\, x \,|\, Gx - F\theta^* \leq w \Leftrightarrow \theta^* \in \Pi_{\mathbb{R}^s}(\mathscr{P}).$$

As a direct consequence of Proposition 3.1, the dimension s' of the smallest affine subspace that contains K^* can be determined by computing the dimension of the projection $\Pi_{\mathbb{R}^s}(\mathscr{P})$.

Definition 3.1. A true inequality of the polyhedron $\mathscr{C} = \{\xi \in \mathbb{R}^n \,|\, B\xi \leq v\}$ is an inequality $B_i\xi \leq v_i$ such that $\exists\, \bar{\xi} \in \mathscr{C} \,|\, B_i\bar{\xi} < v_i$.

Given an H-representation of a polyhedron $\mathscr{C}$ (i.e., a representation of $\mathscr{C}$ as the intersection of halfspaces), the following simple procedure determines the set I of all the nontrue inequalities of $\mathscr{C}$.

Algorithm 3.1.

Step 1. Set $I \leftarrow \varnothing$ and $\mathscr{M} \leftarrow \{1, \ldots, m\}$.
Step 2. If $\mathscr{M}$ is empty, stop.
Step 3. Let j be first element of $\mathscr{M}$; set $\mathscr{M} \leftarrow \mathscr{M} \setminus \{j\}$.
Step 4. Solve the LP problem

$$\min \quad B_j\xi,$$

$$\text{s.t.} \quad B_i\xi \leq v_i, \qquad \forall i \in \mathscr{M},$$

and let ξ^* be an optimizer.

Step 5. If $B_j\xi^* = v_j$, then set $I \leftarrow I \cup \{j\}$.
Step 6. For each element h of $\mathscr{M}$, if $B_h\xi^* < v_h$, then set $\mathscr{M} \leftarrow \mathscr{M} \setminus \{h\}$.
Step 7. Go to Step 2.

The following algorithm describes a standard procedure to determine the dimension $s' \leq s$ of the minimal affine subspace $\mathscr{K}$ that contains K^*; when $s' < s$, it finds the hyperplanes equations defining $\mathscr{K}$. Without loss of generality, we assume that the set K is full dimensional.

Algorithm 3.2.

Step 1. Discard the true inequalities from the constraints in (4).
Step 2. If no inequality is left, then $\mathscr{K} \leftarrow \mathbb{R}^s$; stop.
Step 3. Let $\mathscr{P}_a \triangleq \{(x,\theta) | G_a x - F_a\theta = w_a\}$ be the affine subspace obtained by collecting the remaining nontrue inequalities.
Step 4. Let $\{u_1, \ldots, u_{k'}\}$ be a basis of the kernel of G_a^T.
Step 5. If $k' = 0$, then $\Pi_{\mathbb{R}^s}(\mathscr{P}_a)$ and by Proposition 3.1 K^* are full dimensional, i.e., $\mathscr{K} \leftarrow \mathbb{R}^s$; else, $\mathscr{K} \leftarrow \{\theta | Z\theta = T\}$, where

$$Z = -\begin{bmatrix} u_1^T \\ \vdots \\ u_{k'}^T \end{bmatrix} F_a, \qquad T = \begin{bmatrix} u_1^T \\ \vdots \\ u_{k'}^T \end{bmatrix} w_a.$$

Step 4 can be computed by a standard singular-value decomposition. When $k' > 0$, from the equations $Z\theta = T$, a simple transformation (e.g., a Gauss reduction) leads to a new set of free parameters s', where $s' = s - \operatorname{rank}(Z)$. From now on, without loss of generality, we assume that $\mathscr{K}$ is the whole space $\mathbb{R}^s$.

3.2. Determining the Critical Regions. In this section, we detail the core of the MPLP algorithm, namely the determination of the critical regions CR_{A_i} within the given polyhedral set K. We assume that there is neither primal nor dual degeneracy in the LP problems. These cases will be treated in Section 3.3. We denote by $x^*: K^* \rightarrow \mathbb{R}^n$ the real-valued optimizer function, where $X^*(\theta) = \{x^*(\theta)\}$. The method uses primal feasibility to derive the H-representation of the critical regions, the slackness conditions to compute the optimizer $x^*(\theta)$, and the dual problem to derive the optimal value function $z^*(\theta)$.

The dual problem of (1) is defined as

$$\max_y \quad z = (w + F\theta)^T y, \tag{5a}$$

$$\text{s.t.} \quad G^T y = c, \tag{5b}$$

$$y \leq 0. \tag{5c}$$

The primal feasibility, dual feasibility, and the slackness conditions for problems (1) and (5) are

$$(\mathrm{PF}) \quad Gx \leq w + F\theta, \tag{6a}$$

$$(\mathrm{DF}) \quad G^T y = c, \qquad y \leq 0, \tag{6b}$$

$$(\mathrm{SC}) \quad (G_j x - w_j - F_j\theta)y_j = 0, \qquad \forall j \in J. \tag{6c}$$

Choose an arbitrary vector of parameters $\theta_0 \in \mathscr{K}$ and solve the primal and dual problems (1) and (5) for $\theta = \theta_0$. Let x_0^* and y_0^* be the optimizers of the primal and dual problem, respectively. The value of x_0^* defines the following optimal partition:

$$A(\theta_0) \triangleq \{j \in J | G_j x_0^* - F_j\theta_0 - w_j = 0\}, \tag{7a}$$

$$NA(\theta_0) \triangleq \{j \in J | G_j x_0^* - F_j\theta_0 - w_j < 0\}, \tag{7b}$$

and consequently the critical region $\mathrm{CR}_{A(\theta_0)}$.

By hypothesis, y_0^* is unique; by definition of critical region, y_0^* remains optimal for all $\theta \in \mathrm{CR}_{A(\theta_0)}$. The value function in $\mathrm{CR}_{A(\theta_0)}$ is

$$z^*(\theta) = (w + F\theta)^T y_0^*, \tag{8}$$

which is an affine function of θ on $\mathrm{CR}_{A(\theta_0)}$ as was stated in Theorem 2.4. Moreover, for the optimal partition (7), the PF condition can be rewritten as

$$G_A x^*(\theta) = w_A + F_A\theta, \tag{9a}$$

$$G_{NA} x^*(\theta) < w_{NA} + F_{NA}\theta. \tag{9b}$$

In the absence of dual degeneracy, the primal optimizer is unique and (9a) can be solved to get the solution $x^*(\theta)$. In fact, Eq. (9a) form a system of l equalities where, in the absence of primal degeneracy, $l = n$ is the number of active constraints. From (9a), it follows that

$$x^*(\theta) = -G_A^{-1} F_A\theta + G_A^{-1} w_A = E\theta + Q, \tag{10}$$

which implies the linearity of x^* with respect to θ over $\mathrm{CR}_{A(\theta_0)}$. From the primal feasibility condition (9b), we get immediately the representation of the critical region $\mathrm{CR}_{A(\theta_0)}$,

$$G_{NA}(E\theta + Q) < w_{NA} + F_{NA}\theta. \tag{11}$$

The closure $\overline{\mathrm{CR}}_{A(\theta_0)}$ of $\mathrm{CR}_{A(\theta_0)}$ is obtained by replacing $<$ by $\leq$ in (11).

Once the critical region $\overline{\mathrm{CR}}_{A(\theta_0)}$ has been defined, the rest of the space $R^{\mathrm{rest}} = K \backslash \overline{\mathrm{CR}}_{A(\theta_0)}$ has to be explored and new critical regions generated. An effective approach for partitioning the rest of the space was proposed in

Ref. 9 and formally proved in Ref. 8. In the following, we report the theorem that justifies such a procedure to characterize the rest of the region R^{rest}; see Ref. 8 for the proof.

Theorem 3.1. Let $Y \subseteq \mathbb{R}^n$ be a polyhedron and let $R_0 \triangleq \{x \in Y | Ax \leq b\}$ be a polyhedral subset of Y, where $b \in \mathbb{R}^{m \times 1}$, $R_0 \neq \emptyset$. Also, let

$$R_i = \left\{ x \in Y \,\middle|\, \begin{array}{l} A^i x > b^i \\ A^j x \leq b^j, \ \forall j < i \end{array} \right\}, \qquad i = 1, \ldots, m,$$

and let

$$R^{\text{rest}} \triangleq \bigcup_{i=1}^{m} r_i.$$

Then:

(i) $R^{\text{rest}} \cup R_0 = Y$,

(ii) $R_0 \cap R_i = \emptyset, \qquad R_i \cap R_j = \emptyset, \qquad \forall i \neq j;$

i.e., $\{R_0, R_1, \ldots, R_m\}$ is a partition of Y.

Remark 3.1. The procedure proposed in Theorem 3.1 for partitioning the set of parameters allows one to explore recursively the parameter space; see Remark 3.7 below. Such an iterative procedure terminates after a finite time, as the number of possible combinations of active constraints decreases at each iteration. However, this partitioning strategy defines new polyhedral regions R_k to be explored that are not related to the critical regions which still need to be determined. This may split some of the critical regions, due to the artificial cuts induced by Theorem 3.1. In Ref. 8, postprocessing is used to join cut critical regions that where split. Although algorithms exist for convexity recognition and computation of the union of polyhedra (Ref. 18), such postprocessing operation is computationally expensive. Therefore, in our algorithm, the critical region obtained by (11) is not intersected with halfspaces generated by Theorem 3.1, which is used only to drive the exploration of the parameter space. As a result, no postprocessing is needed to join subpartitioned critical regions. On the other hand, some critical regions may appear more than once. Duplicates can be identified uniquely by the set of active constraints and can be eliminated easily. To this aim, in the implementation of the algorithm, we keep a list of all the critical regions which have been generated already in order to avoid duplicates.

3.3. Degeneracy.

Primal Degeneracy. By applying a Gauss reduction to (9a), we obtain

$$\begin{bmatrix} U & P \\ 0 & D \end{bmatrix} \begin{bmatrix} x \\ \theta \end{bmatrix} = \begin{bmatrix} q \\ r \end{bmatrix}. \tag{12}$$

Assuming that U is nonsingular (the case $\det U = 0$, corresponding to dual degeneracy, will be addressed later), the optimizer is given by

$$x^*(\theta) = -U^{-1}P\theta + U^{-1}q = E\theta + Q, \tag{13}$$

and in (8) one may choose any one of the dual optimizers y_0^* in order to characterize the value function. The H-representation of the critical region $\mathrm{CR}_{A(\theta_o)}$ is

$$G_{NA}(E\theta + Q) < w_{NA} + F_{NA}\theta, \tag{14a}$$

$$D\theta = r. \tag{14b}$$

We distinguish two cases.

Case 1. D is the null matrix and r is the null vector. Then, we have a full-dimensional primal degenerate critical region $\mathrm{CR}_{A(\theta_o)}$.

Case 2. The rank of D is $p > 0$. Then, $\mathrm{CR}_{A(\theta_o)}$ has dimension $s' - p < s' = \dim(\mathscr{K})$. By Theorem 2.1 and the minimality of the dimension s' of $\mathscr{K}$ determined by Algorithm 3.1. we conclude that $\mathrm{CR}_{A(\theta_o)}$ is an $(s' - p)$-dimensional face of another critical region $\mathrm{CR}_{A'}$ for some combination $A' \supset A(\theta_0)$.

Remark 3.2. Case 2 occurs only if the chosen parameter vector θ_0 lies on the face of two or more neighboring critical regions, while Case 1 occurs when a full-dimensional set of parameters makes the LP problem (1) primal degenerate.

Remark 3.3. If Case 2 occurs, to avoid further recursions of the algorithm not producing any full-dimensional critical region (therefore, to avoid lengthening the number of steps required to determine the solution to the MPLP), we perturb the parameter θ_0 by a random vector $\varepsilon \in \mathbb{R}^s$, where

$$\|\varepsilon\|_2 < \min_i \{|S_i\theta_o - q_i| / \sqrt{S_i S_i'}\}, \tag{15}$$

$\|\cdot\|_2$ denotes the standard Euclidean norm and $R_k = \{\theta | S\theta \leq q\}$ is the polyhedral region where we are looking for a new critical region. Equation (15) ensures that the perturbed vector $\theta_0 = \theta_0 + \varepsilon$ is still contained in R_k.

Example 3.1. Consider the MPLP problem

$$\min \quad x_1+x_2+x_3+x_4, \tag{16a}$$

$$\text{s.t.} \quad -x_1+x_5\leq 0, \tag{16b}$$

$$-x_1-x_5\leq 0, \tag{16c}$$

$$-x_2+x_6\leq 0, \tag{16d}$$

$$-x_2-x_6\leq 0, \tag{16e}$$

$$-x_3\leq \theta_1+\theta_2, \tag{16f}$$

$$-x_3-x_5\leq \theta_2, \tag{16g}$$

$$-x_3\leq -\theta_1-\theta_2, \tag{16h}$$

$$-x_3+x_5\leq -\theta_2, \tag{16i}$$

$$-x_4-x_5\leq \theta_1+2\theta_2, \tag{16j}$$

$$-x_4-x_5-x_6\leq \theta_2, \tag{16k}$$

$$-x_4+x_5\leq -\theta_1-2\theta_2, \tag{16l}$$

$$-x_4+x_5+x_6\leq -\theta_2, \tag{16m}$$

$$x_5\leq 1, \tag{16n}$$

$$-x_5\leq 1, \tag{16o}$$

$$x_6\leq 1, \tag{16p}$$

$$-x_6\leq 1, \tag{16q}$$

where

$$K=\{[\theta_1,\theta_2] \mid -2.5\leq \theta_1\leq 2.5, -2.5\leq \theta_2\leq 2.5\}.$$

A possible solution to the MPLP problem (the MPLP is also dual degenerate) is shown in Fig. 1 and the constraints which are active in each associated critical region are reported in Table 1. Clearly, as $x\in\mathbb{R}^6$, CR6 and CR11 are primal degenerate full-dimensional critical regions.

Dual Degeneracy. If dual degeneracy occurs, the set $X^*(\theta)$ may not be a singleton for some $\theta\in K^*$; therefore, the inequalities defining the critical region cannot be determined simply by substitution in (11). In order to get such inequalities, one possibility is to project the polyhedron defined by the equality and inequality (9) onto the parameter space (for efficient tools to compute the projection of a polyhedron, see e.g. Ref. 19), which however does not allow directly defining an optimizer $x^*(\cdot)$. In order to compute

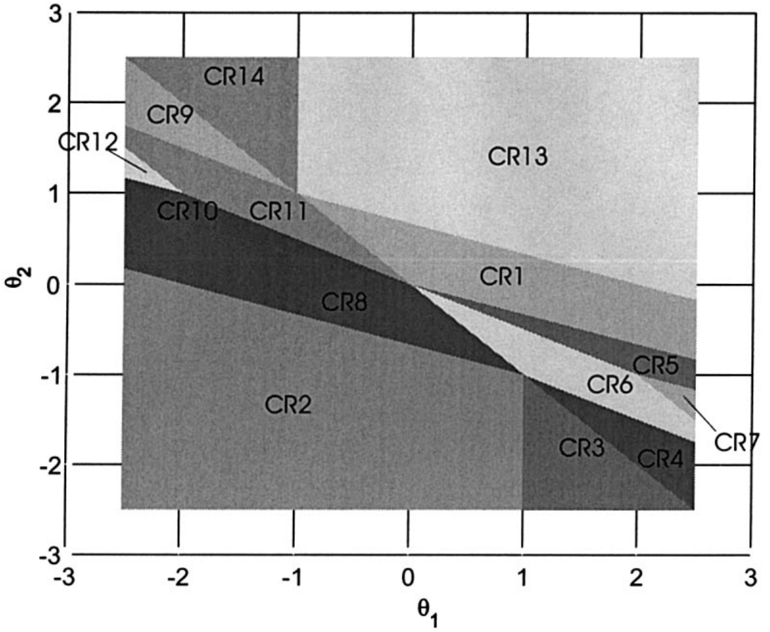


Fig. 1. Polyhedral partition of the parameter space corresponding to the solution of Example 3.1.

Table 1. Critical regions and corresponding optimal value for Example 3.1.

Critical region	Optimal value
$CR1 = CR_{\{2,3,4,7,10,11\}}$	$2x_1 + 3x_2$
$CR2 = CR_{\{1,3,4,5,9,13\}}$	$-2x_1 - 3x_2$
$CR3 = CR_{\{1,3,4,6,9,13\}}$	$-x_1 - 3x_2 - 1$
$CR4 = CR_{\{1,3,6,9,10,13\}}$	$-2x_2 - 1$
$CR5 = CR_{\{1,2,3,7,10,11\}}$	x_1
$CR6 = CR_{\{1,3,6,7,9,10,11,12\}}$	x_1
$CR7 = CR_{\{1,3,7,10,11,15\}}$	x_1
$CR8 = CR_{\{1,3,4,5,9,12\}}$	$-2x_1 - 3x_2$
$CR9 = CR_{\{2,4,8,11,12,14\}}$	$2x_2 - 1$
$CR10 = CR_{\{1,2,4,5,9,12\}}$	$-x_1$
$CR11 = CR_{\{2,4,5,8,9,10,11,12\}}$	$-x_1$
$CR12 = CR_{\{2,4,5,9,12,16\}}$	$-x_1$
$CR13 = CR_{\{2,3,4,7,11,14\}}$	$2x_1 + 3x_2$
$CR14 = CR_{\{2,3,4,8,11,14\}}$	$x_1 + 3x_2 - 1$

one $x^*(\theta) \in X^*(\theta)$ for all θ belonging to a dual degenerate region $\mathrm{CR}_{A(\theta_0)}$, one can choose simply a particular optimizer on a vertex of the feasible set of (1), determine the set $\bar{A}(\theta_0)$ of active constraints for which $G_{\bar{A}(\theta_0)}$ is of full rank, and compute a subset $\overline{\mathrm{CR}}_{\bar{A}(\theta_0)}$ of the dual degenerate critical region (namely, the subset of parameters θ such that only the constraints $\bar{A}(\theta_0)$ are active at the optimizer, which is not a critical region in the sense of Definition 2.3). The algorithm proceeds by exploring the space surrounding $\overline{\mathrm{CR}}_{\bar{A}(\theta_0)}$ as usual. The arbitrariness in choosing an optimizer leads to different ways of partitioning $\mathrm{CR}_{A(\theta_0)}$, where the partitions can be calculated simply from (10) and (11) and in general may overlap. Nevertheless, in each region, a unique optimizer is defined.

This procedure is illustrated in the following example.

Example 3.2. Consider the following MPLP reported in Ref. 2, page 152:

$$\min \quad -2x_1 - x_2, \tag{17a}$$

$$\text{s.t.} \quad x_1 + 3x_2 \leq 9 - 2\theta_1 + \theta_2, \tag{17b}$$

$$2x_1 + x_2 \leq 8 + \theta_1 - 2\theta_2, \tag{17c}$$

$$x_1 \leq 4 + \theta_1 + \theta_2, \tag{17d}$$

$$-x_1 \leq 0, \tag{17e}$$

$$-x_2 \leq 0, \tag{17f}$$

where

$$K = \{[\theta_1, \theta_2] \mid -10 \leq \theta_1 \leq 10, -10 \leq \theta_2 \leq 10\}.$$

The solution is represented in Fig. 2 and the critical regions are listed in Table 2.

The critical region $\mathrm{CR}_{\{2\}}$ is related to a dual degenerate solution with multiple optima. The analytical expression of $\mathrm{CR}_{\{2\}}$ is obtained by projecting the polyhedron

$$x_1 + 3x_2 + 2\theta_1 - \theta_2 < 9, \tag{18a}$$

$$2x_1 + x_2 - \theta_1 + 2\theta_2 = 8, \tag{18b}$$

$$x_1 - \theta_1 - \theta_2 < 4, \tag{18c}$$

$$-x_1 < 0, \tag{18d}$$

$$-x_2 < 0 \tag{18e}$$

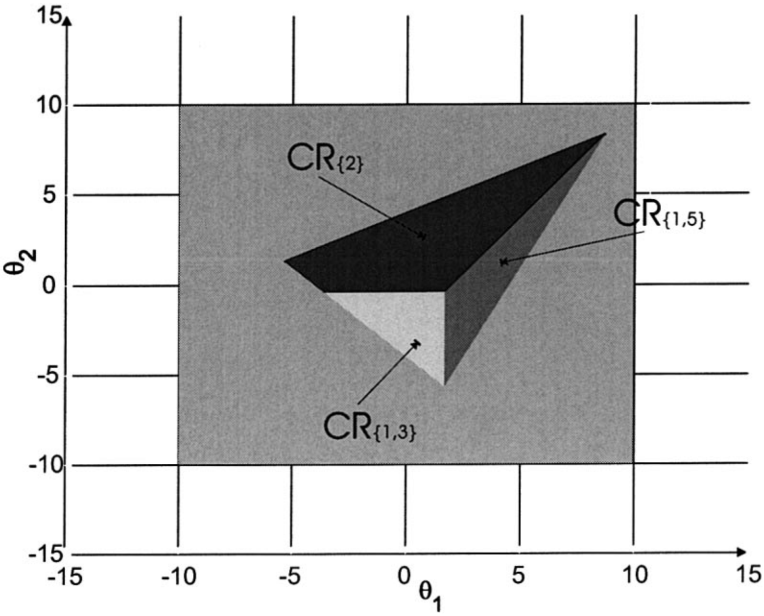


Fig. 2. Polyhedral partition of the parameter space corresponding to the solution of Example 3.2.

Table 2. Critical regions and corresponding optimal value for Example 3.2.

Critical region	Optimizer	Optimal value
$CR_{\{2\}}$	Not single valued	$-2\theta_1+2\theta_2-8$
$CR_{\{1,5\}}$	$x_1^*=-2\theta_1+\theta_2+9,\quad x_2^*=0$	$4\theta_1-2\theta_2-18$
$CR_{\{1,3\}}$	$x_1^*=\theta_1+\theta_2+4,\quad x_2^*=-\theta_1+1.6667$	$-\theta_1-2\theta_2-9.6667$

on the parameter space to obtain

$$CR_{\{2\}}=\left\{\begin{array}{ll}[\theta_1,\theta_2]\,| & 2.5\theta_1-2\theta_2\leq 5\\ & -0.5\theta_1+\theta_2\leq 4\\ & -12\theta_2\leq 5\\ & -\theta_1-\theta_2\leq 4\end{array}\right\}. \tag{19}$$

For all $\theta\in CR_{\{2\}}$, only one constraint is active at the optimum, which makes the optimizer not unique.

Figures 3–7 show two possible ways of covering $CR_{\{2\}}$. The generation of overlapping regions can be avoided by intersecting each new region with

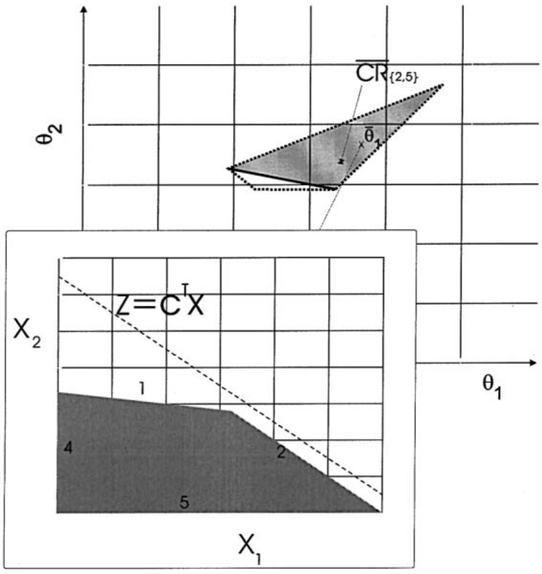


Fig. 3. First region $\overline{\mathrm{CR}}_{\{2,5\}} \subset \mathrm{CR}_{\{2\}}$ and below the feasible set in the x-space corresponding to $\bar{\theta}_1 \in \overline{\mathrm{CR}}_{\{2,5\}}$.

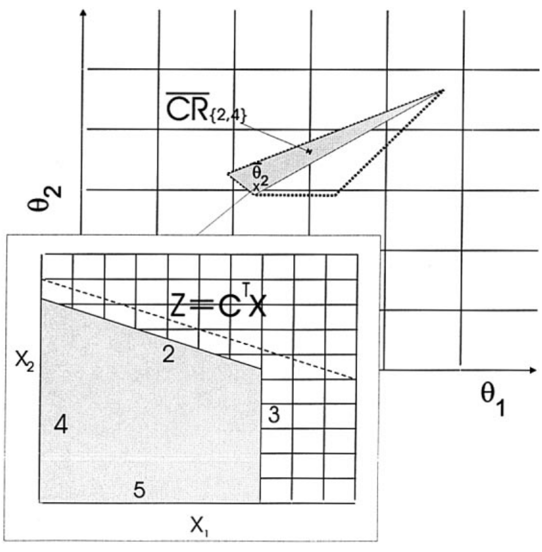


Fig. 4. Second region $\overline{\mathrm{CR}}_{\{2,4\}} \subset \mathrm{CR}_{\{2\}}$ and below the feasible set in the x-space corresponding to $\bar{\theta}_2 \in \overline{\mathrm{CR}}_{\{2,4\}}$.

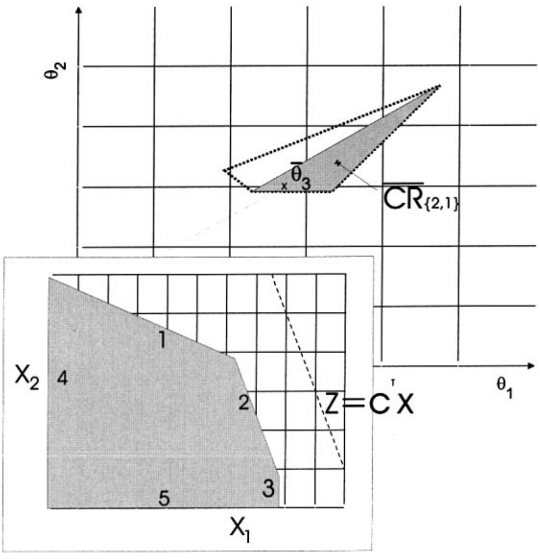


Fig. 5. Third region $\overline{CR}_{\{2,1\}} \subset CR_{\{2\}}$ and below the feasible set in the x-space corresponding to $\bar{\theta}_3 \in \overline{CR}_{\{2,1\}}$.

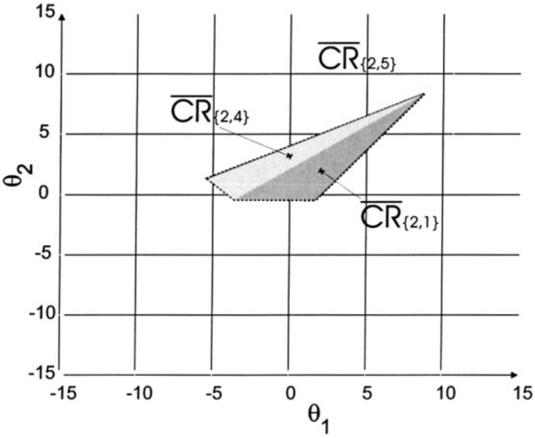
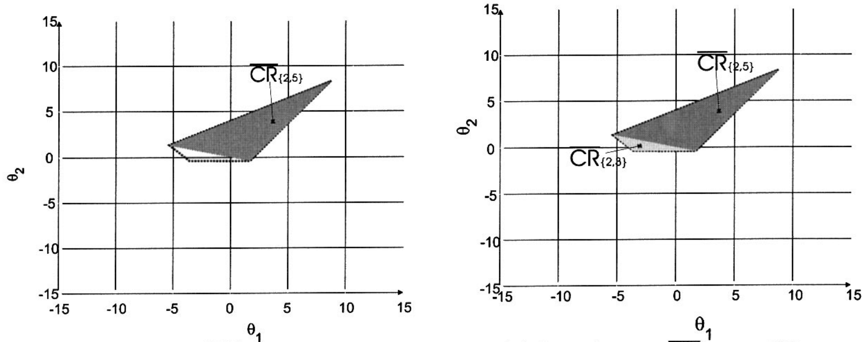


Fig. 6. Final partition of $CR_{\{2\}}$. Note that the region $\overline{CR}_{\{2,5\}}$ is hidden by region $\overline{CR}_{\{2,4\}}$ and region $\overline{CR}_{\{2,1\}}$.



(a) First region $\overline{CR}_{\{2,5\}} \subset CR_{\{2\}}$ (b) Second region $\overline{CR}_{\{2,3\}} \subset CR_{\{2\}}$

Fig. 7. A possible solution to Example 3.2 where the regions $\overline{CR}_{\{2,5\}}$ and $\overline{CR}_{\{2,3\}}$ are nonoverlapping.

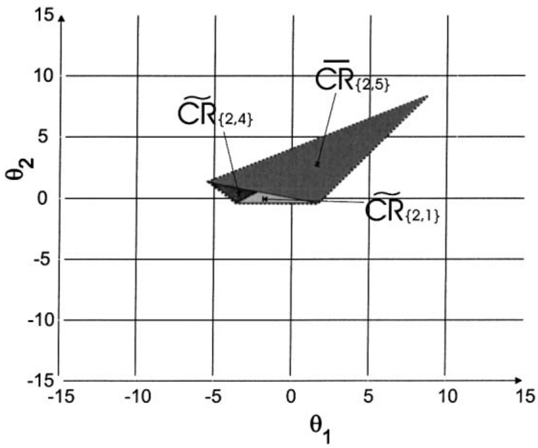


Fig. 8. A possible solution for Example 3.2: $\overline{CR}_{\{2,4\}}$ is obtained by intersecting $\overline{CR}_{\{2,4\}}$ with the complement of $\overline{CR}_{\{2,5\}}$ and $\overline{CR}_{\{2,1\}}$ is obtained by intersecting $\overline{CR}_{\{2,1\}}$ with the complement of $\overline{CR}_{\{2,5\}}$ and $\overline{CR}_{\{2,4\}}$.

the current partition computed so far, as shown in Fig. 8, where $\overline{CR}_{\{2,4\}}$, $\overline{CR}_{\{2,1\}}$ represent the intersected critical regions.

Remark 3.4. In case of dual degeneracy, the method of Ref. 2 explores one of several possible paths on the graph of the bases. Here, the degenerate region is split into nonoverlapping subregions, where an optimizer is uniquely defined. As in Ref. 2, the particular choice of the optimal vertices determines the way of partitioning the degenerate critical region. As a result, the method of Ref. 2 provides a continuous mapping $x^*(\theta)$, while our method may lead to discontinuities over the artificial cuts inside the dual degenerate critical region in case overlapping regions are found. In Figs. 3–6, the regions are overlapping; in Fig. 8, artificial cuts are introduced, with consequent discontinuity of $x^*(\theta)$ at the boundaries inside the degenerate critical region $\mathrm{CR}_{\{2\}}$. On the contrary, no artificial cuts are introduced in Fig. 7. Therefore, the mapping $x^*(\theta)$ is continuous over $\mathrm{CR}_{\{2\}}$.

3.4. Summary of the MPLP Algorithm. Based on the above discussion, the multiparametric LP solver can be summarized in the following recursive Algorithm 3.3.

Algorithm 3.3.

Step 1. Let $Y_k \leftarrow K$ be the current region to be explored and let θ_0 be in the interior of Y_k.
Step 2. Solve the LP (1) and (5) for $\theta = \theta_0$.
Step 3. If the optimizer is not unique, then let x_0^* be one of the optimal vertices of the LP (1) for $\theta = \theta_0$.
Step 4. Let $A(\theta_0)$ be set of active constraints at x_0^* as in (7).
Step 5. If there is primal degeneracy, then let U, P, D be the matrices in (12) after a Gauss reduction to (9a); determine $x^*(\theta)$ from (13) and $\mathrm{CR}_{A(\theta_0)}$ from (14); choose y_0^* among one of the possible optimizers.
Step 6. If there is no primal degeneracy, then determine $x^*(\theta)$ from (10) and $\mathrm{CR}_{A(\theta_0)}$ from (11).
Step 7. Let $z^*(\theta)$ be as in (8) for $\theta = \theta_0$.
Step 8. Partition the rest of the region as in Theorem 3.1 and for each nonempty element R_i of the partition set $Y_k \leftarrow R_i$; go to Step 1.

Remark 3.5. As remarked in Section 3.3, if $\mathrm{rank}(D) > 0$ in Step 5, the region $\mathrm{CR}_{A(\theta_0)}$ is not full dimensional. To avoid further recursion in the algorithm which does not produce any full-dimensional critical region, after computing U, P, D if $D \neq 0$, one should compute a random vector $\varepsilon \in \mathbb{R}^s$ satisfying (15) and such that the LP (1) is feasible for $\theta_0 + \varepsilon$ and then repeat Step 4 with $\theta_0 \leftarrow \theta_0 + \varepsilon$.

Remark 3.6. Note that Step 3 can be executed easily by using an active set method for solving the LP (1) for $\theta = \theta_0$. Note also that primal basic solutions are needed only to define the optimizers in a dual degenerate critical region.

As remarked in the previous section, if one is interested only in characterizing the dual degenerate critical region, without characterizing one of the possible optimizer function $x^*(\cdot)$, Step 3 can be avoided; instead of Step 5 or 6, one can compute the projection $\mathrm{CR}_{A(\theta_0)}$ of the polyhedron (9) on K; note that $A(\theta_0)$ has to be the set of active constraints as defined in Definition 2.3.

Remark 3.7. The algorithm determines the partition of K recursively. After the first critical region is found, the rest of the region in K is partitioned into polyhedral sets $\{R_i\}$ as in Theorem 3.1. By using the same method, each set R_i is further partitioned, and so on. This procedure can be represented on a search tree, with maximum depth equal to the number of combinations of active constraints. The complexity analysis of the algorithm can be found in Ref. 13.

4. Application to Optimal Control Problems

4.1. Finite-Horizon Optimal Control. Consider the following finite-horizon optimal control problem:

$$\min_{u(0),\ldots,u(N-1)} \|Px(N)\|_p + \sum_{k=0}^{N-1} (\|Qx(k)\|_p + \|Ru(k)\|_p), \tag{20}$$

subject to the deterministic linear discrete-time dynamical system

$$x(k+1) = Ax(k) + Bu(k), \qquad k = 0, \ldots, N-1, \tag{21}$$

$$x(0) = x_0, \tag{22}$$

with polyhedral constraints

$$x(k) \in \mathscr{X} \subset \mathbb{R}^n, \qquad k = 1, \ldots, N, \tag{23}$$

$$u(k) \in \mathscr{U} \subset \mathbb{R}^m, \qquad k = 0, \ldots, N-1. \tag{24}$$

We aim at finding the solution $\{u^*(0), \ldots, u^*(N-1)\}$ to problem (20)–(24) in the form of a state-feedback law

$$u^*(k) = f_k(x^*(k)), \qquad k = 0, \ldots, N-1,$$

where

$$x^*(k+1) = Ax^*(k) + Bu^*(k), \qquad x^*(0) = x_0.$$

In other words, we look for the functions $f_k: \mathbb{R}^n \to \mathbb{R}^m$ that, to any given optimal state $x^*(k)$, assign the corresponding optimal control move $u^*(k)$.

In case the sets $\mathscr{X}$, $\mathscr{U}$ are polyhedra and the p–norm in (20) is the one norm or the infinity norm, problem (20)–(24) can be translated into the linear program (Ref. 10)

$$\min_{\xi_0} \quad c_0^T \xi_0, \tag{25a}$$

$$\text{s.t.} \quad G_0 \xi_0 \leq w_0 + F_0 x^*(0), \tag{25b}$$

with

$$\xi_0 \triangleq \{\epsilon_1, \ldots, \epsilon_{N_\epsilon}, u(0), \ldots, u(N-1)\},$$

where ϵ_j, $j = 1, \ldots, N_\epsilon$, are slack variables and N_ϵ depends on the norm used, the number of states x and inputs u, and the control horizon N; see Refs 13–14 for details.

For any given initial condition $x^*(0)$, the linear program (25) provides the corresponding optimal input profile. By considering the initial state $x^*(0)$ as a vector of parameters inside $\mathscr{X}$, the multiparametric solution of (25) provides the optimal profile $u^*(0), \ldots, u^*(N-1)$ as an affine function of the initial state $x(0)$. In particular, $u^*(0)$ as a function of $x^*(0)$,

$$u^*(0) = f_0(x^*(0)), \qquad \forall x^*(0) \in \mathscr{X}, \tag{26}$$

is the desired state feedback control law at time $t = 0$.

Then, consider the same optimal control problem over the reduced time horizon $[i, N]$,

$$\min_{u(i), \ldots, u(N)} \|Px(N)\|_p + \sum_{k=i}^{N-1} (\|Qx(k)\|_p + \|Ru(k)\|_p), \tag{27}$$

with

$$x(k+1) = Ax(k) + Bu(k), \qquad k = i, \ldots, N-1,$$

$$x(i) = x^*(i),$$

and subject to the constraints

$$x(k) \in \mathscr{X} \subset \mathbb{R}^n, \qquad k = i+1, \ldots, N,$$

$$u(k) \in \mathscr{U} \subset \mathbb{R}^m, \qquad k = i, \ldots, N-1.$$

The problem can be translated into the multiparametric LP

$$\min_{\xi_i} \quad c_i^T \xi_i, \tag{28a}$$

$$\text{s.t.} \quad G_i \xi_i \leq w_i + F_i x^*(i), \tag{28b}$$

where ξ_i has suitable dimension.

The first component of the multiparametric solution of (28) has the affine state-feedback form

$$u^*(i) = f_i(x^*(i)), \qquad \forall x^*(i) \in \mathscr{X}. \tag{29}$$

Therefore, the feedback solution

$$u^*(k) = f_k(x^*(k)), \qquad k = 0, \ldots, N-1,$$

of problem (20)–(24) is obtained by solving N multiparametric LP problems.

Example 4.1. Consider the discrete-time system

$$x(t+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t). \tag{30}$$

We want to bring the state of the system as close as possible to the origin $x = [0, 0]$, while minimizing the performance measure

$$\sum_{k=0}^{3} \left(\left\| \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_{k+1} \right\|_\infty + |0.8\, u_k| \right), \tag{31}$$

subject to the input constraints

$$-1 \leq u_k \leq 1, \qquad k = 0, \ldots, 3,$$

and the state constraints

$$-10 \leq x_k \leq 10, \qquad k = 1, \ldots, 4.$$

This task is addressed as shown above. The optimal feedback solution $u^*(0), \ldots, u^*(3)$ is computed in less than one minute on a Pentium II-500

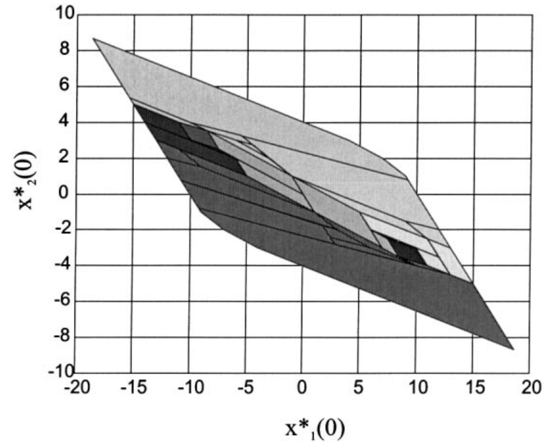


Fig. 9. Partition of the state space for the affine control law $u^*(0)$ of Example 4.1.

MHz running Matlab 5.3 and the Nag toolbox 1.3 by solving four multi-parametric-LP problems and the corresponding polyhedral partition of the state-space is depicted in Figs. 9–12.

5. Conclusions

We presented a novel algorithm for solving multiparametric linear programs, which follows a geometric approach based on the direct exploration

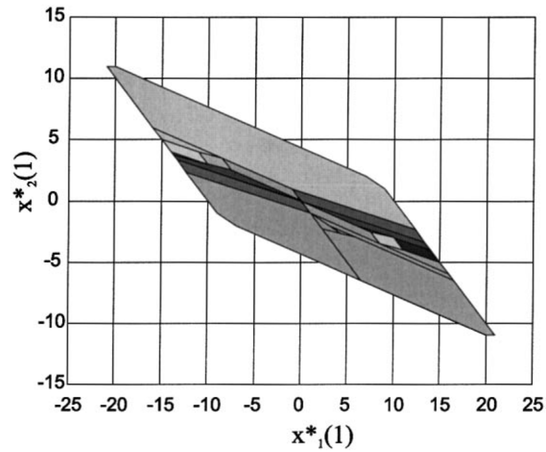


Fig. 10. Partition of the state space for the affine control law $u^*(1)$ of Example 4.1.

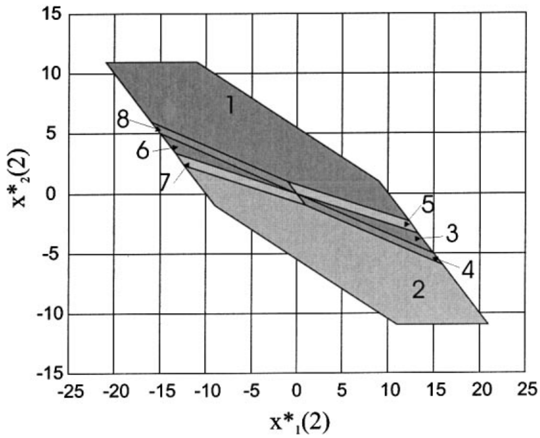


Fig. 11. Partition of the state space for the affine control law $u^*(2)$ of Example 4.1.

of the parameter space, rather than visiting different bases of the associated LP tableau. The resulting algorithm was implemented and tested on several examples, which show the feasibility of the method. The algorithm allows computing the explicit feedback control law of finite-time constrained optimal control problems for discrete-time linear systems and was applied

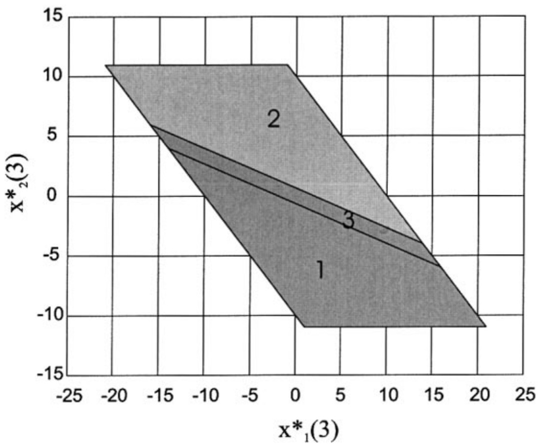


Fig. 12. Partition of the state space for the affine control law $u^*(3)$ of Example 4.1.

successfully to solve optimal control design for a range of problems such as traction control (Ref. 22). Current research is devoted to mixed-integer multiparametric programming in order to solve optimal control problems for systems containing both dynamical and logic components (Ref. 14).

References

1. Gass, S., and Saaty, T., *The Computational Algorithm for the Parametric Objective Function*, Naval Research Logistics Quarterly, Vol. 2, pp. 39–45, 1955.
2. Gal, T., *Postoptimal Analyses, Parametric Programming, and Related Topics*, 2nd Edition, de Gruyter, Berlin, Germany, 1995.
3. Gal, T., and Greenberg, H., *Advances in Sensitivity Analysis and Parametric Programming*, International Series in Operations Research and Management Science, Kluwer Academic Publishers, Dordrecht, Netherlands, Vol. 6, 1997.
4. Gal, T., and Nedoma, J., *Multiparametric Linear Programming*, Management Science, Vol. 18, pp. 406–442, 1972.
5. Filippi, C., *On the Geometry of Optimal Partition Sets in Multiparametric Linear Programming*, Technical Report 12, Department of Pure and Applied Mathematics, University of Padova, Padova, Italy, 1997.
6. Schechter, M., *Polyhedral Functions and Multiparametric Linear Programming*, Journal of Optimization Theory and Applications, Vol. 53, pp. 269–280, 1987.
7. Fiacco, A. V., *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*, Academic Press, London, UK, 1983.
8. Bemporad, A., Morari, M., Dua, V., and Pistikopoulos, E., *The Explicit Linear-Quadratic Regulator for Constrained Systems*, Automatica, Vol. 38, pp. 3–20, 2002.
9. Dua, V., and Pistikopoulos, E., *An Algorithm for the Solution of Multiparametric Mixed Integer Linear Programming Problems*, Annals of Operations Research, Vol. 99, pp. 123–139, 2000.
10. Zadeh, L., and Whalen, L., *On Optimal Control and Linear Programming*, IRE Transaction on Automatic Control, Vol. 7, pp. 45-46, 1962.
11. Morari, M., and Lee, J., *Model Predictive Control: Past, Present, and Future*, Computers and Chemical Engineering, Vol. 23, pp. 667–682, 1999.
12. Mayne, D., Rawlings, J., Rao, C., and Scokaert, P., *Constrained Model Predictive Control: Stability and Optimality*, Automatica, Vol. 36, pp. 789–814, 2000.
13. Bemporad, A., Borrelli, F., and Morari, M., *Model Predictive Control Based on Linear Programming: The Explicit Solution*, IEEE Transactions on Automatic Control, Vol. 47, pp. 1974–1985, 2002.
14. Borrelli, F., *Discrete-Time Constrained Optimal Control*, PhD Thesis, Automatic Control Labotatory, ETH, Zurich, Switzerland, 2002.
15. Adler, I., and Monteiro, R., *A Geometric View of Parametric Linear Programming*, Algorithmica, Vol. 8, pp. 161–176, 1992.

16. Mehrotra, S., and Monteiro, R., *Parametric and Range Analysis for Interior-Point Methods.*, Technical Report, Department of Systems and Industrial Engineering, University of Arizona, Tucson, Arizona, 1992.
17. Berkelaar, A., Roos, K., and Terlaky, T., *The Optimal Set and Optimal Partition Approach to Linear and Quadratic Programming*, Advances in Sensitivity Analysis and Parametric Programming, Edited by T. Gal and H. Greenberg, International Series in Operations Research and Management Science, Kluwer Academic Publishers, Dordrecht, Netherlands, Vol. 6, 1997.
18. Bemporad, A., Fukuda, K., and Torrisi, F., *Convexity Recognition of the Union of Polyhedra*, Computational Geometry, Vol. 18, pp. 141–154, 2001.
19. Fukuda, K., *cdd/cdd+ Reference Manual*, Institute for Operations Research, ETH-Zentrum, Zurich, Switzerland, 0.61 (cdd) and 0.75 (cdd+) editions, 1997.
20. Murty, K. G., *Computational Complexity of Parametric Linear Programming*, Mathematical Programming, Vol. 19, pp. 213–219, 1980.
21. Goodman, J., and O'Rourke, J., *Handbook of Discrete and Computational Geometry*, Discrete Mathematics and Its Applications, CRC Press, New York, NY, 1997.
22. Borrelli, F., Bemporad, A., Fodor, M., and Hrovat, D., *A Hybrid Approach to Traction Control*, Hybrid Systems: Computation and Control, Edited by A. Sangiovanni-Vincentelli and M. D. Benedetto, Lecture Notes in Computer Science, Springer Verlag, Berlin, Germany, Vol. 2034, pp. 162–174, 2001.