Proceedings of the 42nd IEEE
Conference on Decision and Control
Maui, Hawaii USA, December 2003

Approximate Convex Multiparametric Programming

Alberto Bemporad[1] and Carlo Filippi[2]

Abstract

For convex multiparametric nonlinear programming problems we propose a recursive algorithm for approximating, within a given suboptimality tolerance, the value function and an optimizer as functions of the parameters. The approximate solution is expressed as a piecewise affine function over a simplicial partition of a subset of the feasible parameters, and it is organized over a tree structure for efficiency of evaluation. The case of multiparametric semidefinite programming is examined and exemplified on a test example. The approach opens up the application of explicit receding horizon techniques to several robust model predictive control schemes based on convex optimization, such as linear matrix inequalities.

1 Introduction

Parametric programming considers optimization problems where the data depend on one or more parameters. Parametric programming techniques systematically subdivide the parameter space into characteristic regions where the optimal value and an optimizer are given as explicit functions of the parameters.

In recent years, a new interest in parametric programming arose in the model predictive control (MPC) community. MPC is a well-known technique widely used in the process industry for the automatic regulation of plants under operating constraints [7, 15]. In model predictive control, the next command action is obtained by solving an optimization problem where the cost function and the constraints depend on the current sensor measurements. In the classic setting, the optimization problem is solved on-line at each time step. However, most of the optimization effort may be moved off-line by solving a multiparametric program where variables correspond to command inputs, and parameters correspond to sensor measurements [4, 17, 1].

A vast literature is concerned with parametric programming, but it is almost always restricted to a single parameter and/or to very well known problems, like linear programs [10, 5] or convex quadratic programs [4, 18, 17]. We may distinguish two main issues explaining these limitations of the research efforts: (*i*) contrarily to the case of one scalar parameter, parametric solutions with a vector parameter are difficult to analyze by a human decision maker; (*ii*) for more general convex optimization problems (e.g., semidefinite programming), the exact shape of the optimal value function may be unknown.

Designing methods to get an approximate description of the optimal value function and of a sub-optimal solution is a promising direction for coping with the above issues. A seminal contribution in this direction was given by Fiacco [8, Chapter 9]. In the context of general convex parametric nonlinear programming, he sketched a strategy for approximating optimal value functions along a mono-dimensional cut of the parameter space. Essentially, Fiacco noted that optimal primal solutions associated with two fixed parameter vectors may be used to compute an affine upper bound along the line segment joining the same parameter vectors; furthermore, optimal dual solutions associated with the two parameter vectors may be used to compute a piecewise affine lower bound along the same line segment. By following similar observations, Filippi [9] developed an algorithm for approximate multiparametric linear programming. A completely different approach was used by Bemporad and Filippi [3] to get an approximate solution to a multi-parametric strictly convex quadratic programming problem. They proposed to enlarge the exact characteristic region corresponding to a fixed active constraint set by relaxing the first-order optimality conditions, while preserving primal feasibility. Another approach was taken by Johansen [13] for obtaining piecewise affine approximate solutions of multiparametric nonlinear programming problems using local quadratic approximations.

In this paper we consider a quite general class of multiparametric convex programs, and propose a recursive algorithm for approximating, within a given prescribed tolerance, the value function and an optimizer as explicit functions of the parameters. Our approach is inspired by the lines suggested in Fiacco [8, Chapter 9] and Filippi [9], and its main ideas are the following: (*i*) given a full-dimensional simplex in the parameter space and an optimizer for each simplex vertex, the linear interpolation of the given solutions gives a primal feasible approximation of an optimizer inside the simplex; (*ii*) if the resulting absolute error in the objective exceeds a prescribed tolerance then the simplex is split into smaller simplices where it applies recursively; (*iii*) initial simplices are obtained by a triangularization of a polyhedral estimate of the set of feasible parameters. The resulting approximate solution is expressed as a piecewise affine function over a simplicial partition of a subset of the set of feasible parameters, and organized over a tree structure for efficiency of evaluation (a similar tree structure based on boxes rather than simplices was used in [14] to obtain approximate solutions to

[1] Alberto Bemporad is with Dip. Ingegneria dell'Informazione, Università di Siena, Italy, `http://www.dii.unisi.it/~bemporad`

[2] Carlo Filippi is with Dip. Matematica Pura e Applicata, Università di Padova, Italy, `http://www.math.unipd.it/~carlo`

multiparametric quadratic programs). The algorithm applies to the general framework of convex multiparametric programming, and may conveniently fitted for special cases of relevant interest. In particular, the case of multiparametric semidefinite programming is briefly examined and exemplified on a test example.

The goal of our approach is to open up the application of explicit receding horizon techniques to several robust model predictive control schemes based on convex optimization.

2 Convex Multiparametric Programming

Consider the convex multiparametric program

$$(CP_\theta) \qquad \begin{array}{ll} \min_x & f(x,\theta) \\ \text{s. t.} & g_i(x,\theta) \le 0 \quad (i=1,\dots,p) \\ & Ax + B\theta + d = 0 \end{array}$$

where $x \in \mathbb{R}^n$ are the decision variables, $\theta \in \mathbb{R}^m$ are the parameters, $f : \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}$ is the objective function, $g_i : \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}$, for all $i = 1,\dots,p$, A is a $q \times n$ real matrix, B is a $q \times m$ real matrix, and $d \in \mathbb{R}^q$. We assume that f and g_i $(i = 1,\dots,p)$ are jointly convex in both the variables and the parameters. Clearly, the left-hand side of each equality constraint is jointly affine in both the variables and the parameters. We are interested in characterizing the solution of problem (CP_θ) for a given polytopic set of parameters

$$\Theta = \{\theta \in \mathbb{R}^m : \ Q\theta \le R\} \subset \mathbb{R}^m.$$

The solution of problem (CP_θ) is defined as follows. The *feasible parameter set* Θ_f is the set of all $\theta \in \Theta$ for which the corresponding problem (CP_θ) admits a solution, i.e., there exists a vector x satisfying the constraints of (CP_θ). The *value function* $V^\star : \Theta_f \mapsto \mathbb{R}$ is the function that associates with every $\theta \in \Theta_f$ the corresponding unique optimal value of (CP_θ). The *optimizer set function* $X^\star : \Theta_f \mapsto 2^{\mathbb{R}^n}$ is the function that associates to a parameter vector $\theta \in \Theta_f$ the corresponding set of optimizers $X^*(\theta) = \{x \in \mathbb{R}^n : f(x,\theta) = V^\star(\theta)\}$ of problem (CP_θ). An *optimizer function* $x^\star : \Theta_f \mapsto \mathbb{R}^n$ is a function that associates to a parameter vector $\theta \in \Theta_f$ (one of) the optimizer(s) $x^\star(\theta) \in X^\star(\theta)$.

The following basic result for convex multiparametric programming was proved in [16, Lemma 1] in the absence of equality constraints:

Lemma 1 *Consider the multiparametric problem (CP_θ) and let f, g_i be jointly convex functions of (x,θ), for all $i = 1,\dots,p$. Then, Θ_f is a convex set and $V^\star$ is a convex function of θ.*

The result can be easily generalized to the presence of linear equality constraints.

Hereafter we assume that Θ_f and Θ are full-dimensional sets. A numerical test for verifying such an assumption will be provided in Section 4.

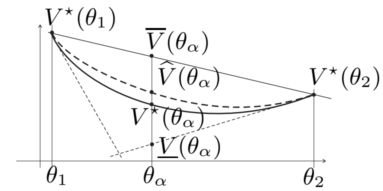


Figure 1: Approximation of the value function in convex parametric programming: the scalar case

3 Approximate Multiparametric Programming

3.1 Upper Bounds on the Value Function

Let $\theta^0, \theta^1, \dots, \theta^m \in \mathbb{R}^m$ be affinely independent points in Θ_f, and define S as the following m-dimensional simplex:

$$S \triangleq \{\theta \in \mathbb{R}^m : \theta = \textstyle\sum_{k=0}^m \mu_k \theta^k, \ \sum_{k=0}^m \mu_k = 1, \ \mu_k \ge 0 \ (k = 0,1,\dots,m)\}. \qquad (1)$$

Let x^k be an optimizer of (CP_{θ^k}), for all $k = 0,1,\dots,m$; define the matrices

$$M \triangleq \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \theta^0 & \theta^1 & \cdots & \theta^m \end{bmatrix}, \ X \triangleq \begin{bmatrix} x^0 & x^1 & \cdots & x^m \end{bmatrix}, \qquad (2)$$

and note that by construction M is nonsingular. A proof of the following simple result can be found in [9].

Proposition 1 *The system of linear inequalities $M^{-1}\binom{1}{\theta} \ge 0$ is a minimal representation of S.*

In the following, we introduce upper and lower bounds on $V^\star$ inside S. Such bounds generalize to the multidimensional case the concepts introduced by Fiacco [8, Chapter 9] to bound the value function of a parametric convex program inside a line segment (cf. also [13]).

Define the vector $v \triangleq [V^\star(\theta^0) \ V^\star(\theta^1) \ \cdots \ V^\star(\theta^m)]'$, and the function

$$\widehat{x}(\theta) \triangleq XM^{-1}\begin{pmatrix}1\\ \theta\end{pmatrix}. \qquad (3)$$

Furthermore, define

$$\widehat{V}(\theta) \triangleq f(\widehat{x}(\theta),\theta), \qquad (4)$$

and $\overline{V}(\theta) \triangleq v'M^{-1}\binom{1}{\theta}$. Note that both $\widehat{x}$ and $\overline{V}$ are affine functions of θ.

Proposition 2 *For all $\theta \in S$ the vector $\widehat{x}(\theta)$ is a feasible solution of (CP_θ), and*

$$\overline{V}(\theta) \ge \widehat{V}(\theta) \ge V^\star(\theta). \qquad (5)$$

Proof: We first prove that $\widehat{x}(\theta)$ is feasible. The vector $\mu = M^{-1}\binom{1}{\theta}$ is the unique solution of $M\mu = \binom{1}{\theta}$. Thus, if $\theta \in S$ then $\mu \ge 0$, $\sum_{k=0}^m \mu_k = 1$, $\theta = \sum_{k=0}^m \mu_k\theta^k$, and $\widehat{x}(\theta) = \sum_{k=0}^m \mu_k x^k$. As a consequence, for all $i = 1,\dots,p$, $g_i(\widehat{x}(\theta),\theta) = g_i(\sum_{k=0}^m \mu_k x^k, \sum_{k=0}^m \mu_k\theta^k) \le \sum_{k=0}^m \mu_k g_i(x^k,\theta^k) \le 0$, where the first inequality follows from the joint convexity of g_i with respect to x and θ. Furthermore, $A\widehat{x}(\theta) + B\theta + d = \sum_{k=0}^{m} \mu_k(Ax^k + B\theta^k + d) = 0$. To prove (5), we note that $\overline{V}(\theta) = \sum_{k=0}^{m} \mu_k V^\star(\theta^k) = \sum_{k=0}^{m} \mu_k f(x^k, \theta^k) \geq f(\sum_{k=0}^{m} \mu_k x^k, \sum_{k=0}^{m} \mu_k \theta^k) = f(\widehat{x}(\theta), \theta) = \widehat{V}(\theta)$, where the inequality follows from the joint convexity of f, and $\widehat{V}(\theta) = f(\widehat{x}(\theta), \theta) \geq f(x^\star(\theta), \theta) = V^\star(\theta)$, where $x^\star(\theta)$ denote any optimizer of (CP_θ). □

Thus, $\overline{V}$ and $\widehat{V}$ are both upper bounds of $V^\star$ on S, tight on every vertex on S, where $\overline{V}$ is affine in θ and $\widehat{V}$ is tighter than $\overline{V}$.

3.2 Lower Bounds on the Value Function

Assuming that a subgradient of $V^\star$ is available at every vertex of S, we can construct a piecewise affine lower bound of $V^\star$. More precisely, let s^k be a subgradient of $V^\star$ at θ^k $(k = 0, 1, \ldots, m)$. Since $V^\star$ is convex, we have $V^\star(\theta) \geq V^\star(\theta^k) + (s^k)'(\theta - \theta^k)$. As a consequence, we define $\underline{V}(\theta) \triangleq \max_{k=0,1,\ldots,m}\{V^\star(\theta^k) + (s^k)'(\theta - \theta^k)\}$. By construction,

$$\underline{V}(\theta) \leq V^\star(\theta) \quad \text{for all } \theta \in S, \tag{6}$$

so that $\underline{V}$ is a piecewise affine lower bound on $V^\star$ inside S, tight at every vertex of the simplex.

We assume the following:

Assumption 1 *Functions f and g_i $(i = 1, \ldots, p)$ are differentiable with respect to both x and θ inside their domain.*

For convenience, let $g(x, \theta) \triangleq [g_1(x, \theta), \ldots, g_p(x, \theta)]'$. The Karush-Kuhn-Tucker optimality conditions for problem (CP_θ) are (see, e.g., [6, Chapter 5]):

$$\begin{array}{l} g(x,\theta) \leq 0, \quad Ax + B\theta + d = 0, \\ \lambda \geq 0, \quad \lambda' g(x,\theta) = 0, \\ \nabla_x f(x,\theta) + J_x g(x,\theta)'\lambda + A'\nu = 0, \end{array} \tag{7}$$

where $\lambda \in \mathbb{R}^p$ and $\nu \in \mathbb{R}^q$ are the vectors of Lagrange multipliers, $\nabla_x f(x, \theta) \in \mathbb{R}^n$ denotes the gradient of f with respect to x, and $J_x g(x, \theta)$ denotes the $p \times n$ Jacobian matrix of the partial derivatives of g with respect to x.

Proposition 3 *Let (x^k, λ^k, ν^k) be a solution of (7) for $\theta = \theta^k$, for any $k = 0, 1, \ldots, m$. Then*

$$s^k \triangleq \nabla_\theta f(x^k, \theta^k) + J_\theta g(x^k, \theta^k)'\lambda^k + B'\nu^k$$

is a subgradient of $V^\star$ of (CP_θ) at θ^k, where $\nabla_\theta f(x, \theta) \in \mathbb{R}^m$ denotes the gradient of f with respect to θ and $J_\theta g(x, \theta)$ denotes the $p \times m$ Jacobian matrix of partial derivatives of g with respect to θ.

Proof: See [2]. □

A similar result was shown by Fiacco [8, Chapter 9] using an auxiliary lower-bounding multiparametric linear programming problem.

In case a primal-dual method is used for computing $V^\star(\theta^k)$, both optimal primal variables x^k and Lagrange multipliers λ^k, ν^k are available. If also the derivatives of f and g_i are available, then a subgradient s^k valid at θ^k, and therefore a linear lower bound on $V^\star$, can be immediately constructed according to Proposition 3.

3.3 Error Estimates Inside a Simplex

We wish to approximate $V^\star$ by using $\widehat{V}$ inside the simplex S, with vertices θ^k, $k = 0, 1, \ldots, m$. In this way, the maximum absolute error we introduce is

$$\epsilon^{MAX}(S) \triangleq \max_\theta\{\widehat{V}(\theta) - V^\star(\theta) : \theta \in S\}.$$

Unfortunately, the above optimization problem is a nonconvex DC programming problem, and thus the exact evaluation of $\epsilon^{MAX}(S)$ is, in general, hard [12]. For this reason, we analyze two practically computable upper bounds on $\epsilon^{MAX}(S)$.

Proposition 4 *Let $s^k \in \mathbb{R}^m$ be a subgradient of $V^\star$ at θ^k, and let $w_k \triangleq -V^\star(\theta^k) - (s^k)'\theta^k$, for all $k = 0, 1, \ldots, m$. Define*

$$\epsilon^{LP}(S) \triangleq \left\{ \begin{array}{ll} \max_x & \overline{V}(\theta) - t \\ \text{s. t.} & (s^k)'\theta - t \leq w_k \quad (k = 0, \ldots, m) \\ & -M^{-1}\binom{1}{\theta} \leq 0, \end{array} \right. \tag{8}$$

where $M = \left[\begin{smallmatrix} 1 & 1 & \cdots & 1 \\ \theta^0 & \theta^1 & \cdots & \theta^m \end{smallmatrix}\right]$, and $\theta^0, \ldots, \theta^m$ are the vertices of S. Then, $V^\star(\theta) \geq \widehat{V}(\theta) - \epsilon^{LP}(S)$, $\forall \theta \in S$.

Proof: We have $\max_\theta\{\widehat{V}(\theta) - V^\star(\theta) : \theta \in S\} \leq \max_\theta\{\overline{V}(\theta) - \underline{V}(\theta) : \theta \in S\} = \max_\theta\{\overline{V}(\theta) - \max_k\{V^\star(\theta^k) + (\delta^k)'(\theta - \theta^k) : k = 0, 1, \ldots, m\} : \theta \in S\} = \max_{\theta,t}\{\overline{V}(\theta) - t : t \geq V^\star(\theta^k) + (\delta^k)'(\theta - \theta^k)\ (k = 0, 1, \ldots, m), \theta \in S\} = \epsilon^{LP}(S)$. □

Proposition 5 *Let*

$$\epsilon^{CP}(S) \triangleq \left\{ \begin{array}{ll} \max_{x,\theta} & \overline{V}(\theta) - f(x,\theta) \\ \text{s. t.} & g(x,\theta) \leq 0 \\ & Ax + B\theta + d = 0 \\ & -M^{-1}\binom{1}{\theta} \leq 0, \end{array} \right. \tag{9}$$

where M is defined as in Proposition 4. Then, $V^\star(\theta) \geq \widehat{V}(\theta) - \epsilon^{CP}(S)$, $\forall \theta \in S$.

Proof: Let $F(\theta) \triangleq \{x \in \mathbb{R}^n : g(x, \theta) \leq 0, Ax + B\theta + d = 0\}$ (i.e., $F(\theta)$ is the feasible set of (CP_θ)). We have: $\epsilon^{MAX}(S) = \max_\theta\{\widehat{V}(\theta) - V^\star(\theta) : \theta \in S\} \leq \max_\theta\{\overline{V}(\theta) - V^\star(\theta) : \theta \in S\} = \max_\theta\{\overline{V}(\theta) - \min_x\{f(x, \theta) : x \in F(\theta)\} : \theta \in S\} = \max_\theta\{\overline{V}(\theta) + \max_x\{-f(x, \theta) : x \in F(\theta)\} : \theta \in S\} = \max_{x,\theta}\{\overline{V}(\theta) - f(x, \theta) : x \in F(\theta), \theta \in S\} = \epsilon^{CP}(S)$. □

Proposition 6 *For all simplices $S \subseteq \Theta_f$, $\epsilon^{LP}(S) \geq \epsilon^{CP}(S) \geq \epsilon^{MAX}(S)$.*

Proof: From the proofs of Proposition 4 and Proposition 5, and from (6), we have: $\epsilon^{LP}(S) = \max_\theta\{\overline{V}(\theta) - \underline{V}(\theta) : \theta \in S\} \geq \max_\theta\{\overline{V}(\theta) - V^\star(\theta) : \theta \in S\} = \epsilon^{CP}(S)$, $\forall S \subseteq \Theta_f$. □

Proposition 6 shows that both ϵ^{LP} and ϵ^{CP} are upper bounds on ϵ^{MAX}. Computing ϵ^{LP} involves solving of a linear program with $m + 1$ variables, whereas computing ϵ^{CP} involves solving a convex program with $m + n$ variables. However, obtaining the subgradients used to compute ϵ^{LP} may require an additional effort, unless the parametric program takes some special form. In the next section we provide a recursive approximation algorithm for problem (CP_θ) that only makes use of ϵ^{CP}.

4 An Approximate Multiparametric Solver

We are in a position to state a basic approximation algorithm for (CP_θ). The algorithm consists of two phases. In the initialization phase, we test if the set Θ_f of feasible parameters is full dimensional. If this is the case, a polyhedral inner approximation of the set Θ_f of feasible parameters is obtained and subdivided into a collection of simplices. In the second phase, each simplex is recursively subdivided into smaller simplices until the desired degree of accuracy in approximating the value function is reached in each simplex.

We start by describing the recursive phase. Let S be a full-dimensional simplex contained in Θ_f, defined as in (1), and let $\widehat{x}$ and $\widehat{V}$ be defined as in (3) and (4) respectively. Let $\epsilon > 0$ be a given maximum error. The following Algorithm 4.1 summarizes the recursive steps.

---

1. Build M and X as defined in (2);
2. **if** M is nonsingular **then**
   - 2.1. Solve problem (9), getting $\epsilon^{CP}(S)$ and $(\overline{x}, \overline{\theta})$;
   - 2.2. **if** $\epsilon^{CP}(S) > \epsilon$ **then**
     - 2.2.1. **for** $k = 0, 1, \ldots, m$ **do**
       - 2.2.1.1 Replace the kth vertex of S by $\overline{\theta}$ and let S_k be the new simplex;
       - 2.2.1.2 Call this algorithm on S_k;
   - 2.3. **else** Return M^{-1} and X;

Algorithm 4.1: Recursive splitting of an initial simplex and approximation of the value function and of the optimizer. Note that $\widehat{x}$ and $\widehat{V}$ are readily evaluated by using the returned matrices.

In Algorithm 4.1, we used $\epsilon^{CP}(S)$ for convenience. However, we remark that any of the upper bounds on the error described in the previous section may be used in the same way. Note that, at each recursive iteration, the current simplex is split into at most $m+1$ full-dimensional simplices with nonoverlapping interiors.

Algorithm 4.1 builds up a piecewise affine function $\widehat{x} : S \mapsto \mathbb{R}^n$ and a piecewise analytical function $\widehat{V} : S \mapsto \mathbb{R}$ such that: (i) $\widehat{x}(\theta)$ is a feasible solution of (CP_θ) for all $\theta \in S$, (ii) $\widehat{V}(\theta) = f(\widehat{x}(\theta), \theta)$ for all $\theta \in S$, and (iii) $0 \leq \widehat{V}(\theta) - V^\star(\theta) \leq \epsilon$ for all $\theta \in S$. Note that $\widehat{x}$ and $\widehat{V}$ may not be continuous on the boundary of the returned simplices. As a consequence, the approximate optimizer and value function may be defined more than once for some $\theta \in S$, though this fact can happen only on a subset of S with null measure.

By using $\epsilon^{CP}(S)$ (or even $\epsilon^{LP}(S)$), the proposed method controls the absolute error on the value function with respect to $\overline{V}$, which constitutes an approximation of $V^\star$ worse than the actually returned $\widehat{V}$. As a consequence, there may be cases where a simplex is split because $\epsilon^{CP}(S) > \epsilon$ though the maximum difference between $\widehat{V}$ and $V^\star$ is less than the prescribed ϵ. In order to possibly avoid unnecessary splits, consider the error quantity $\underline{\epsilon}(S) \triangleq \widehat{V}(\overline{\theta}) - V^\star(\overline{\theta}) = f(\widehat{x}(\overline{\theta}), \overline{\theta}) - V^\star(\overline{\theta}) \leq \epsilon^{MAX}(S)$, where $\epsilon^{MAX}(S)$ is the maximum absolute error on S. If $\underline{\epsilon}(S) > \epsilon$ then clearly $\epsilon^{MAX}(S) > \epsilon$ and hence the simplex S must be split. On the other hand, when $\underline{\epsilon}(S) \leq \epsilon$ there is the possibility that the actual error $\epsilon^{MAX}(S)$ is smaller than ϵ. A technique based on a piecewise linear approximation of $\widehat{V}$ over S for estimating $\epsilon^{MAX}(S)$ with an arbitrary precision is described in [2].

Remark 4.1 If in every recursive call vector $\overline{\theta}$ lies in the interior of its simplex, then $\widehat{x}$ and $\widehat{V}$ are both continuous functions of the parameter θ. If the continuity property is required, we may force the above condition by imposing in (9) the tighter constraint $M^{-1}\binom{1}{\theta} \geq \sigma e$, where σ is a comparatively small positive scalar and $e \in \mathbb{R}^{m+1}$ is a vector of ones. This is equivalent to letting $\mu_k \geq \sigma > 0$ for all $k = 0, 1, \ldots, m$, where μ_k are the coefficients of the convex combination of the vertices of the simplex. As an alternative, in order to enforce continuity and obtain a geometric balance, one may always decide to split S in its center $\frac{1}{m+1}\sum_{k=0}^{m} \theta^k$.

4.1 Initialization

So far, we have assumed that Θ_f is a full-dimensional set. This assumption can be verified as follows. First of all, a necessary condition for Θ_f to be full dimensional is that the equality constraints $Ax + B\theta = d$ do not restrict θ to lie on a lower-dimensional affine subspace of $\mathbb{R}^m$ (i.e., the set $\{\theta \in \mathbb{R}^m : \exists x \in \mathbb{R}^n : Ax + B\theta + d = 0\}$ has dimension m). This can be easily verified by computing a Gauss reduction of $[A\ B\ d]$ and then checking if equality constraints of the form $a'\theta = \alpha$ appear with $a \neq 0 \in \mathbb{R}^m$. Assuming that the linear constraints $Ax + B\theta + d = 0$ do not reduce the dimension of Θ_f, let $S(\theta, \rho) = \mathrm{conv}(\theta + \rho e^0, \theta + \rho e^1, \ldots, \theta + \rho e^m)$, where "conv" denotes the convex hull, e^j is the jth column of the m-by-m identity matrix, $j = 1, \ldots, m$, and $e^0 = 0 \in \mathbb{R}^m$. We determine the largest simplex $S(\theta, \rho)$ contained in Θ_f, by solving

$$\begin{array}{rl} \max\limits_{\theta, \rho, y^0, \ldots, y^m} & \rho \\ \text{s. t.} & g(y^k, \theta + \rho e^k) \leq 0, \\ & Ay^k + B(\theta + \rho e^k) = d, \ (k = 0, \ldots, m) \\ & Q(\theta + \rho e^k) \leq R \end{array} \tag{10}$$

which is a convex program in $(m+1)(n+1)$ variables. Then Θ_f is full-dimensional if and only if the optimal value ρ^* is strictly positive (the volume of the largest simplex being $(\rho^*)^m/m! > 0$).

Once the full-dimensionality of Θ_f is tested, we determine an inner polyhedral approximation $\widehat{\Theta}_f$ through a "ray-shooting" procedure, described as follows. Let $r^0, r^1 \ldots, r^{t+m}$ be $m+t+1$ directions in $\mathbb{R}^m$, $t \geq 0$, such that the convex positive cone $C = \{\theta \in \mathbb{R}^m : \theta = \sum_{i=0}^{m+t} \mu_i r^i,\ \mu_i \geq 0\} = \mathbb{R}^m$. For instance, r^i may be obtained by collecting uniformly distributed samples of the unit hyper-sphere. For each $i = 0,\ 1,\ \ldots,\ m+t$, solve the convex problem $\max\limits_{x,\theta}\{(r^i)'\theta : g(x, \theta) \leq 0, Ax + B\theta + d = 0, Q\theta \leq R\}$, and let (x^i, θ^i) be the obtained optimal solution. Set $\widehat{\Theta}_f \triangleq \text{conv}(\theta^0, \theta^1, \dots, \theta^{m+t})$, and discard redundant[1] vectors θ^i. For simplicity of notation, we assume that $\theta^{m+h+1}, \dots, \theta^{m+t}$ are the redundant vertices, where $h \leq t$ and $h \geq 0$ because Θ_f is full dimensional. So, let $\theta^0, \theta^1, \dots, \theta^{m+h}$ be the vertices of $\widehat{\Theta}_f$, and assume that they are lexicographically ordered, i.e., $\theta^0 \leq \theta^1 \leq \dots \leq \theta^{m+h}$ (componentwise inequalities). Rather than computing a hyperplane representation of $\widehat{\Theta}_f$, we compute a set of simplices $S_1, \dots, S_h$ such that *(i)* $\cup_{i=1}^h S_i = \widehat{\Theta}_f$, and *(ii)* S_i, S_j have mutually disjoint interiors for $i \neq j$. The simplices S_i are defined recursively as follows: (1) Let $L_0 \triangleq \{\theta^0, \theta^1, \dots, \theta^m\}$ be the set of the first $m+1$ vertices of $\widehat{\Theta}_f$, according to the lexicographic order; set $S_0 \triangleq \text{conv}(L_0)$; (2) for all $j = 1, \dots, h$: Let $\widetilde{\theta}$ be the (unique) element in L_{j-1} such that $\widetilde{\theta} + \beta(\theta^{m+j} - \widetilde{\theta}) \in S_{j-1}$ for some $\beta > 0$. Set $L_j \triangleq L_{j-1} \setminus \{\widetilde{\theta}\} \cup \{\theta^{m+j}\}$; set $S_j \triangleq \text{conv}(L_j)$. More efficient ways of obtaining the triangularization $S_0, S_1, \dots, S_h$ of $\widehat{\Theta}_f$ may be devised, although this is beyond the scope of this paper.

Note that the full-dimensionality test (10) may be substituted by the condition $\text{rank}\left[\begin{smallmatrix} 1 & 1 & \cdots & 1 \\ \theta^0 & \theta^1 & \cdots & \theta^{m+h}\end{smallmatrix}\right] = m$, i.e., by testing that $\widehat{\Theta}_f$ is a full-dimensional polyhedron. On the other hand, test (10) is independent on the choice of the directions r^i, which provides more numerical robustness.

4.2 Evaluation of the Solution

Algorithm 4.1 provides the solution of (CP_θ) organized on a tree structure T. The root node of T corresponds to the given set of parameters of interest $\Theta = \{\theta : Q\theta \leq R\}$. At the first level, the nodes correspond to the initial simplices $S_0, S_1, \dots, S_h$ obtained by the ray-shooting procedure. Each node at the first level is the root of a subtree corresponding to the simplicial partition produced by the recursive procedure.

The multiparametric solution is defined over the simplices associated with the leaf nodes, and in principle the internal nodes do not provide any information. However, by keeping such an information, the tree can be exploited to evaluate the multiparametric solution in a very efficient manner. In fact, it is easy to check that for a given $\theta \in \mathbb{R}^m$, determining the simplex which contains θ requires at most $m^2(h+(N-1)(m+1))$ basic arithmetic operations, where N is the depth of T, and h is the number of simplices S_i obtained by the ray-shooting procedure. Note that this way of evaluating the solution requires not only the storage of (M^{-1}, X) in the leaf nodes, where M, X are defined in (2), but also the storage of M^{-1} in all the internal nodes.

[1]A vector θ^i is redundant if $\text{conv}(\theta^0, \dots, \theta^{m+t}) = \text{conv}(\theta^0, \dots, \theta^{i-1}, \theta^{i+1}, \dots, \theta^{m+t})$. Redundancy can be easily tested via linear programming.

5 Approximate Multiparametric Semidefinite Programming

Parametric semidefinite programming (SDP) was addressed in [11] for the case of scalar perturbations of the cost function. In order to deal with SDP problems with multiparametric perturbations, the analysis and the algorithm developed in the previous sections for the convex multiparametric program (CP_θ) can be extended to generalized inequalities and generalized convexity. Here we focus on a parametric semidefinite program where all functions are affine and the inequalities are defined with respect to the proper cone $\mathbb{S}_+^p$ of symmetric positive semidefinite $p \times p$ real matrices; we denote the condition $P \in \mathbb{S}_+^p$ by $P \succcurlyeq 0$.

More precisely, we formulate a multiparametric semidefinite programming problem as follows:

$$\begin{array}{ll} \min & c'x + f'\theta \\ \text{s. t.} & \sum_{i=1}^n x_i F_i + G_0 + \sum_{j=1}^m \theta_j G_j \succcurlyeq 0 \\ & Ax + B\theta + d = 0 \end{array} \tag{11}$$

where $c \in \mathbb{R}^n$, $f \in \mathbb{R}^m$, F_i are real symmetric $p \times p$ matrices for all $i = 1, \dots, n$, G_j are real symmetric $p \times p$ matrices for all $j = 0, 1, \dots, m$, $A \in \mathbb{R}^{q \times n}$, $B \in \mathbb{R}^{q \times m}$, and $d \in \mathbb{R}^q$. Note that the term $f'\theta$ in the objective function is irrelevant for the optimization.

Lemma 2 *Let Θ_f be the feasible parameter set and let $V^\star$ be the value function of problem (11). Then, Θ_f is a convex set and $V^\star$ is a convex function.*

Proof: See [2]. □

As the convexity of $V^\star$ is the key hypothesis behind our development, Lemma 2 implies that the analysis of Section 3 and the solver of Section 4 can be extended to a problem of the form (11) in a straightforward manner.

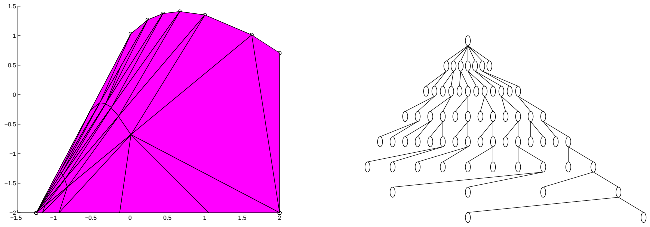
5.1 A Numerical Example

Consider the multiparametric semidefinite program

$$\begin{array}{l} \min\limits_{x \in \mathbb{R}^3} \; x_1 - 2x_2 + x_3 \\ \text{s. t.} \; \begin{bmatrix} 1 & 2 & -3 \\ 2 & 4 & -1 \\ -3 & -1 & 3 \end{bmatrix} + \begin{bmatrix} 1 & -1 & 2 \\ -1 & 1 & 3 \\ 2 & 3 & 2 \end{bmatrix} \theta_1 + \begin{bmatrix} -1 & 1 & 0 \\ 1 & 1 & 2 \\ 0 & 2 & -2 \end{bmatrix} \theta_2 + \\ \begin{bmatrix} 3 & -2 & 4 \\ -2 & 1 & -2 \\ 4 & -2 & -2 \end{bmatrix} x_1 + \begin{bmatrix} -3 & 1 & 1 \\ 1 & -2 & -1 \\ 1 & -1 & 1 \end{bmatrix} x_2 + \begin{bmatrix} 5 & 4 & 2 \\ 4 & 1 & 1 \\ 2 & 1 & -1 \end{bmatrix} x_3 \succcurlyeq 0. \end{array} \tag{12}$$

We are interested in approximating the multiparamentric solution within the box $\Theta = \{\theta \in \mathbb{R}^2 : \; -2 \leq \theta_1, \theta_2 \leq 2\}$ with a precision $\epsilon = 0.5$. To this end, we run Algorithm 4.1, which returns the solution after 2.05 s[2]. In Figure 2(a) we depict the simplicial partition determined by the algorithm, while in Figure 2(b) the associated tree structure for evaluation of the approximate solution, which consists of eight levels. The polyhedral partition in Figure 2(a) contains 35 regions, corresponding to the leaf nodes in Figure 2(b). In Figure 3 we show the value function $V^\star(\theta)$ and the error

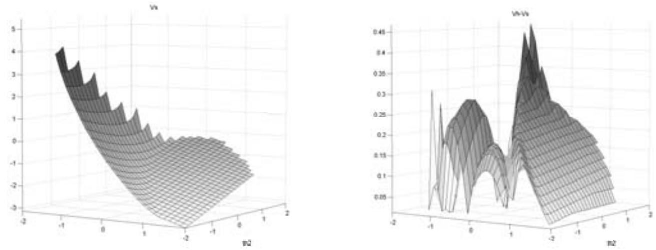
[2]The results were obtained on a PC Pentium III mobile 850 Mhz running Matlab 5.3 and the SDP solver [19].



(a) Partition in θ-space. Ray-shoots for estimating the set of feasible parameters are represented by circles

(b) Tree structure for evaluation of the approximate solution (total number of nodes: 64)

Figure 2: Approximate multiparametric solution of problem (12)



(a) Value function $V^\star(\theta)$

(b) Error $\widehat{V}(\theta) - V^\star(\theta)$

Figure 3: Multiparametric solution associated with problem (12)

$\widehat{V}(\theta) - V^\star(\theta)$, where $V^\star(\theta)$ was computed numerically by gridding. Note that the error is always smaller than the prescribed precision $\epsilon = 0.5$, is zero at the vertices of the simplices, and is always below about 10% of the range of values of the optimal value function.

Note that in the present multiparametric SDP context only an approximate description of an optimal solution may be obtained, as an exact analytical characterization of the value function $V^\star$ is not yet known. This is a topic currently under investigation.

6 Conclusions

In this paper we have provided a recursive algorithm for determining approximate multiparametric solutions of convex nonlinear programming problems, where the value function is approximated within a given suboptimality threshold. The approximate solution is expressed as a piecewise affine function over a simplicial partition of a given set of feasible parameters.

We envision several applications of the technique, especially for the practical implementation of robust model predictive control schemes based on convex optimization, of which several formulations are already available in the literature. It is a topic for further research to analyze which schemes lead to multiparametric programs that are convex both in the variables and in the parameters, and how to maintain robust stability properties in spite of the approximation error.

The results of this paper were also extended for approximating solutions to multiparametric geometric programs within a given relative error.

References

[1] A. Bemporad, F. Borrelli, and M. Morari. Model predictive control based on linear programming — The explicit solution. *IEEE Trans. Automatic Control*, 47(12):1974–1985, December 2002.

[2] A. Bemporad and C. Filippi. Approximate multiparametric convex programming. Technical Report 07/03, Dip. Ingegneria dell'Informazione, University of Siena, 2003. `http://www.dii.unisi.it/~bemporad`.

[3] A. Bemporad and C. Filippi. Suboptimal explicit RHC via approximate multiparametric quadratic programming. *Journal of Optimization Theory and Applications*, 117(1):9–38, April 2003.

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

[5] F. Borrelli, A. Bemporad, and M. Morari. A geometric algorithm for multi-parametric linear programming. *Journal of Optimization Theory and Applications*, 2002. Accepted for publication as a regular paper.

[6] S. Boyd and L. Vandenberghe. Convex optimization. `http://www.stanford.edu/~boyd/cvxbook.html`, December 2002.

[7] Model Predictive Control. *E.F. Camacho and C. Bordons.* Advanced Textbooks in Control and Signal Processing. Springer-Verlag, London, 1999.

[8] A.V. Fiacco. *Introduction to sensitivity and stability analysis in nonlinear programming.* Academic Press, London, U.K., 1983.

[9] C. Filippi. An algorithm for approximate multiparametric linear programming. *Journal of Optimization Theory and Applications.* To appear.

[10] T. Gal. *Postoptimal Analyses, Parametric Programming, and Related Topics.* de Gruyter, Berlin, 2nd edition, 1995.

[11] D. Goldfarb and K. Scheinberg. On parametric semidefinite programming. *Applied Numerical Mathematics*, 29:361–377, 1999.

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

[13] T.A. Johansen. On multi-parametric nonlinear programming and explicit nonlinear model predictive control. In *Proc. 41th IEEE Conf. on Decision and Control*, pages 2768–2773, Las Vegas, Nevada, USA, December 2002.

[14] T.A. Johansen and A. Grancharova. Approximate explicit constrained linear model predictive control via orthogonal search tree. *IEEE Trans. Automatic Control*, 58(5):810–815, May 2003.

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

[16] O.L. Mangasarian and J.B. Rosen. Inequalities for stochastic nonlinear programming problems. *Operations Research*, 12:143–154, 1964.

[17] M.M. Seron, J.A. DeDoná, and G.C. Goodwin. Global analytical model predictive control with input constraints. In *Proc. 39th IEEE Conf. on Decision and Control*, pages 154–159, 2000.

[18] P. Tøndel, T.A. Johansen, and A. Bemporad. An algorithm for multi-parametric quadratic programming and explicit MPC solutions. *Automatica*, 2003. To appear.

[19] L. Vandenberghe, S. Boyd, and B. Alkire. SP — Software for semidefinite programming (Version 1.1). `http://www.ee.ucla.edu/~vandenbe/sp.html`, May 1999.