Mathematical Programming 44 (1989) 247-269
North-Holland



# AN EXTENSION OF THE SIMPLEX ALGORITHM FOR SEMI-INFINITE LINEAR PROGRAMMING

E.J. ANDERSON and A.S. LEWIS

*Engineering Department, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK*

Received 18 March 1986
Revised manuscript received 13 April 1987

We present a primal method for the solution of the semi-infinite linear programming problem with constraint index set $S$. We begin with a detailed treatment of the case when $S$ is a closed line interval in $\mathbb{R}$. A characterization of the extreme points of the feasible set is given, together with a purification algorithm which constructs an extreme point from any initial feasible solution. The set of points in $S$ where the constraints are active is crucial to the development we give. In the non-degenerate case, the descent step for the new algorithm takes one of two forms: either an active point is dropped, or an active point is perturbed to the left or right. We also discuss the form of the algorithm when the extreme point solution is degenerate, and in the general case when the constraint index set lies in $\mathbb{R}^p$. The method has associated with it some numerical difficulties which are at present unresolved. Hence it is primarily of interest in the theoretical context of infinite-dimensional extensions of the simplex algorithm.

*Key words*: Linear programs, semi-infinite programs, extreme points, simplex algorithm.

## 1. Introduction

There are a number of well-tried methods available for the solution of semi-infinite programming problems. Hettich [9] gives a review of these methods and a fuller treatment of the whole subject of semi-infinite linear programming can be found in Glashoff and Gustafson [5]. In this paper we describe an algorithm which is markedly different to the usual techniques. Our method works directly with extreme points of the feasible set for the primal semi-infinite linear program. It is in this sense a simplex-like algorithm, and so has considerable intrinsic interest in the context of attempts to extend the simplex algorithm to more general infinite-dimensional linear programs. In particular, a number of authors have considered the possibility of a continuous time simplex method (see Perold [14] and the references therein). Any such method must be able to deal effectively with the semi-infinite problems we investigate in this paper, since these are special cases of the general continuous time linear program. Thus our investigation into the difficulties inherent in the construction of a simplex-like algorithm for semi-infinite linear programming is relevant to a much broader class of problems.

One of the aims in extending the simplex algorithm to infinite-dimensional linear programs is to avoid making an explicit discretization of the problem. Our degree of success in achieving this in the semi-infinite case is reported in this paper. It is

however inevitable that some discretization is used in the numerical implementation of the algorithm, for instance in checking that a trial solution is feasible for the problem.

In addition to the algorithm's theoretical interest, it avoids some of the difficulties encountered by standard solution techniques. The resolution of the numerical problems raised by the implementation of the algorithm could thus prove to be of some practical interest. Moreover, the new method serves as a powerful illustration of the approach to general linear programming problems described by Nash [13].

We consider the semi-infinite program when the index set of the constraints, $S$, is taken as a polyhedral subset of $\mathbb{R}^p$. The semi-infinite program then has the form

$$\begin{aligned} \text{SIP1:} \quad & \text{minimize} && c^{\mathrm{T}}x \\ & \text{subject to} && a(s)^{\mathrm{T}}x \geqslant b(s) \text{ for all } s \in S, \\ & && x \in \mathbb{R}^n, \end{aligned}$$

where $a$ and $b$ are continuous functions from $\mathbb{R}^p$ to $\mathbb{R}^n$ and $\mathbb{R}$ respectively. A dual problem for SIP1 can be formulated as follows:

$$\begin{aligned} \text{SIP1}^*\text{:} \quad & \text{maximize} && \int_S b(s)\, \mathrm{d}\omega(s) \\ & \text{subject to} && \int_S a(s)\, \mathrm{d}\omega(s) = c, \\ & && \omega \geqslant 0, \ \omega \in M[S], \end{aligned}$$

where $M[S]$ is the space of regular Borel measures on $S$ (see Rudin [16]).

The algorithm described here is a primal algorithm; it approaches the optimal solution through a sequence of solutions each of which are feasible for SIP1. Consequently we shall not make any direct use of the dual problem and the exact form in which it is posed will not be important. A key element in our approach is an analysis of the extreme point structure of the primal problem. The plan of the paper is as follows. We begin by giving in the next section some fundamental definitions and results for the general abstract linear program, which we shall later specialize to the semi-infinite case. A characterization of the extreme points is given in Section 3 for the case $S = [0, 1] \subset \mathbb{R}$. In Section 4 we show how an improved extreme point solution can be obtained from any feasible solution, and in the following section an optimality check is given. This optimality check is the basis of the improvement step described in Section 6, for the non-degenerate case. Degeneracy is an important phenomenon for the semi-infinite program, and we discuss in Section 7 the structure of the feasible set near a degenerate extreme point. This leads to a method for making improvement steps in the degenerate case. We finish by describing the form that the algorithm takes when $S$ is a polyhedral subset of $\mathbb{R}^p$, and by discussing the relationship of this new method with more well-known techniques.

The algorithm that we describe here has been introduced in outline in Anderson [1]. This present paper completes the description given there, and adds to it a treatment of the degenerate case and the case where the constraint index set is of more than one dimension. We have yet to make a full-scale implementation of the algorithm, so that we are only able to give limited numerical results.

## 2. The general linear program: Results and definitions

We begin by reviewing the framework for general linear programming described by Nash [13]. Let $X$ and $Z$ be real vector spaces and $X_+$ a convex cone in $X$. $X_+$ is called the *positive cone* in $X$ and defines a partial order "$\geqslant$" on $X$ by

$$x \geqslant y \text{ for } x, y \in X \quad \text{if and only if} \quad x - y \in X_+.$$

We write $\theta$ for the zero element of a vector space, so that for $x \in X$, $x \geqslant \theta$ if and only if $x \in X_+$. For $c^* \in X^*$, the dual of $X$, denote the image of $x$ under $c^*$ by $\langle x, c^* \rangle$. Let $A: X \to Z$ be a linear map, and let $b \in Z$. We consider the linear program

$$\begin{aligned} \text{LP:} \quad & \text{minimize} \quad \langle x, c^* \rangle \\ & \text{subject to} \quad Ax = b, \\ & \qquad\qquad x \geqslant \theta, \ x \in X. \end{aligned}$$

The *feasible region* of LP is the set $\{x \in X: Ax = b \text{ and } x \geqslant \theta\}$ and $\xi \in X$ is called an *extreme point* of LP if $\xi$ is an extreme point of the feasible region of LP.

The first result we need gives a simple algebraic characterization of the extreme points of LP. For any $\xi \in X$ we define the following set:

$$B(\xi) = \{x \in X: \exists \lambda > 0, \lambda \in \mathbb{R} \text{ with } \xi + \lambda x \geqslant \theta \text{ and } \xi - \lambda x \geqslant \theta\}.$$

Notice that $B(\xi)$ is a subspace of $X$. We denote the null space of the map $A$ by $N(A)$. The following lemma, due to Nash, is straightforward to establish.

**Lemma 1.** *$\xi \in X$ is an extreme point of* LP *if and only if $B(\xi) \cap N(A) = \{\theta\}$.*

If $\xi$ is an extreme point of LP, we can form the direct sum of $B(\xi)$ and $N(A)$. We denote this subspace by $D(\xi)$. $\xi$ is called *degenerate* if

$$D(\xi) = B(\xi) \oplus N(A) \neq X.$$

It is not hard to check that this definition corresponds to the usual one when the linear program is finite.

The definitions above can be used to establish an important characterization of the optimal extreme points for LP. Suppose that $\xi$ is an extreme point of LP and let $P_{N(A)}: D(\xi) \to N(A)$ be the natural projection. Define the *reduced cost* for $\xi$ to be a map $c_\xi^*: D(\xi) \to \mathbb{R}$ given by

$$\langle x, c_\xi^* \rangle = \langle P_{N(A)}(x), c^* \rangle \quad \text{for } x \in D(\xi).$$

A proof of the following result can be found in Nash [13].

**Lemma 2.** *If $\xi$ is an extreme point of* LP, *then $\xi$ is an optimal solution of* LP *if and only if $c_\xi^*$ is positive on $D(\xi)$, i.e. $\langle x, c_\xi^* \rangle \geq 0$ for each $x \in X_+ \cap D(\xi)$.*

In the next section we show how these definitions and results specialize to the semi-infinite linear program.

## 3. The semi-infinite linear program

We consider the semi-infinite linear program SIP1 with $S = [0, 1] \subset \mathbb{R}$. We will assume that the components $a_1, a_2, \ldots, a_n$, of the function $a$, and the function $b$, are all members of $C^\infty[0, 1]$. In order to put SIP1 into the form LP, we introduce a slack variable $z \in C^\infty[0, 1]$. The problem then becomes

SIP2: minimize $c^T x$

subject to $a(s)^T x - z(s) = b(s)$ for all $s \in [0, 1]$,

$x \in \mathbb{R}^n$, $z \geq 0$, $z \in C^\infty[0, 1]$.

In the notation of Section 2, we have

$$X = \mathbb{R}^n \times C^\infty[0, 1],$$

$$Z = C^\infty[0, 1],$$

$$A(x; z) = a(\cdot)^T x - z(\cdot),$$

$$\langle (x; z), c^* \rangle = c^T x,$$

and

$$X_+ = \{(x; z): z \geq 0\}.$$

Now we shall consider a specific $(\xi; \zeta)$, feasible for SIP2. We call the set $\{s \in S: \zeta(s) = 0\}$ the *active points*. We shall assume that there are only a finite number of active points which we denote by $\{s_1, s_2, \ldots, s_k\}$. At the active points the slack function $\zeta$, which is greater than or equal to zero throughout $[0, 1]$, attains a strict local minimum with value 0. We shall need to keep track of the order of these zeros of $\zeta$ and so we define $d(i)$ to be the smallest non-negative integer $j$ such that $\zeta^{(j+1)}(s_i) \neq 0$. If $s_i$ is in the interior of the line segment $[0, 1]$, then $d(i)$ will be odd. We shall assume that $d(i)$ is defined (i.e. is less than $\infty$) for each $i = 1, 2, \ldots, k$.

We now set about characterizing the extreme points of SIP2 using the results of Section 2.

**Lemma 3.** *The subspace $B(\xi; \zeta)$ is given by*

$$B(\xi; \zeta) = \{(x; z) \in \mathbb{R}^n \times C^\infty[0, 1]: z^{(j)}(s_i) = 0, j = 0, \ldots, d(i), i = 1, \ldots, k\}.$$

**Proof.** It is easy to see that there exists a $\lambda > 0$ with $\zeta + \lambda z \geq 0$ and $\zeta - \lambda z \geq 0$ on $[0, 1]$ if and only if

$$\sup\{|z(s)|/\zeta(s): s \in [0, 1], s \neq s_1, s_2, \ldots, s_k\} < \infty.$$

Now $|z(s)|/\zeta(s)$ is continuous everywhere in $[0, 1]$ except possibly at $s_1, s_2, \ldots, s_k$. By l'Hôpital's rule, $|\lim_{s \to s_i} z(s)/\zeta(s)| < \infty$ if and only if $z^{(j)}(s_i) = 0$, $j = 0, \ldots, d(i)$, $i = 1, \ldots, k$, and this establishes the result. □

Let $m = k + \sum_{i=1}^{k} d(i)$. We define the $m \times n$ matrix $\hat{A}$ by

$$\hat{A} = (a(s_1), a'(s_1), \ldots, a^{(d(1))}(s_1), a(s_2), \ldots, a(s_k), \ldots, a^{(d(k))}(s_k))^{\mathrm{T}}, \quad (1)$$

so that the rows of $\hat{A}$ are the values of $a$ and its derivatives at the active points. We then obtain the following characterization of extreme points:

**Theorem 4.** *$(\xi; \zeta)$ is an extreme point of* SIP2 *if and only if the columns of $\hat{A}$ are linearly independent, or equivalently*

$$\operatorname{span}\{a^{(j)}(s_i): j = 0, \ldots, d(i), i = 1, \ldots, k\} = \mathbb{R}^n.$$

**Proof.** $(x; z) \in B(\xi; \zeta) \cap N(A)$ if and only if $a(s)^{\mathrm{T}}x - z(s) = 0$, for $s \in [0, 1]$, and $z^{(j)}(s_i) = 0$, $j = 0, \ldots, d(i)$, each $i = 1, \ldots, k$. Thus by Lemma 1, $(\xi; \zeta)$ is extreme if and only if

$$\{x: a^{(j)}(s_i)^{\mathrm{T}}x = 0, j = 0, \ldots, d(i), \text{ each } i = 1, \ldots, k\} = \{0\},$$

i.e. if and only if the columns of $\hat{A}$ are linearly independent. □

## 4. Purification

In this section we will consider the problem of how to construct an extreme point of SIP2. This will be a necessary first step in any solution algorithm which is based on extreme points. We will make the following assumption concerning the problem SIP2:

$$\{x: c^{\mathrm{T}}x \leq 0 \text{ and } a(s)^{\mathrm{T}}x \geq 0, s \in [0, 1]\} = \{0\}. \quad (2)$$

Assumption (2) will hold in particular if the feasible region is bounded. Under this assumption we can generate an extreme point of SIP2 by applying the *purification algorithm* described below to any feasible starting point. We shall return to the question of finding an initial feasible solution in Section 6. The algorithm proceeds at each step by moving in such a way as to maintain all the previous zeros of the slack variable, until a new zero is obtained.

0. Take $(\xi^1; \zeta_1)$ feasible for SIP2, and set $r = 1$.

*Iteration r*

1. Let $\{s_1, \ldots, s_k\}$ be the active points corresponding to $(\xi^r; \zeta_r)$, and define $d(i)$ as in Section 3, for $i = 1, \ldots, k$.
2. Define a subspace $T_r \subseteq \mathbb{R}^n$ by setting $T_r = \mathbb{R}^n$ if $k = 0$, and

$$T_r = \{x: a^{(j)}(s_i)^T x = 0, j = 1, \ldots, d(i), \text{ each } i = 1, \ldots, k\},$$

   otherwise. If $T_r = \{0\}$, STOP: $(\xi^r; \zeta_r)$ is an extreme point.
3. Set $g^r = -P_{T_r}(c)$ (where $P_{T_r}$ is the orthogonal projection onto $T_r$). If $g^r = 0$, pick a nonzero $g^r$ arbitrarily in $T_r$.
4. Set $\beta_r = \sup\{-[a(s)^T g^r / \zeta_r(s)]: s \in [0, 1], s \neq s_1, \ldots, s_k\}$.
5. Set $\xi^{r+1} = \xi^r + (1/\beta_r) g^r$, and $\zeta_{r+1}(\cdot) = a(\cdot)^T \xi^{r+1} - b(\cdot)$.
6. Increase $r$ by 1 and return to step 1.

**Theorem 5.** *The above algorithm terminates at an extreme point in at most n iterations. Moreover, the cost at this point is not greater than the cost at the initial point* $(c^T \xi^1)$.

**Proof.** Let us denote $(1/\beta_r)$ by $\alpha_r$. Notice that

$$\alpha_r = \sup\{\alpha: (\xi^r; \zeta_r) + \alpha(g^r; a(\cdot)^T g^r) \text{ is feasible for SIP2}\},$$

and this supremum is attained. By definition, $c^T g^r \leq 0$, so the cost cannot increase at any step, and by assumption (2), $\alpha_r < \infty$. Also notice that

$$B(\xi^r; \zeta_r) \cap N(A) = \{(x; a(\cdot)^T x): x \in T_r\}.$$

By the definition of $B(\xi^r; \zeta_r)$, there is a $\lambda > 0$ such that

$$\zeta_r(s) \pm \lambda a(s) g^r \geq 0 \quad \text{for } s \in [0, 1],$$

so $\alpha_r > 0$. Clearly $T_{r+1} \subseteq T_r$. But by the characterization of $\alpha_r$ given above,

$$(g^r; a(\cdot)^T g^r) \notin B(\xi^{r+1}; \zeta_{r+1}),$$

so $g^r \notin T_{r+1}$. Thus $T_{r+1} \subset T_r$ strictly, and since each $T_r$ is a subspace of $\mathbb{R}^n$, the algorithm terminates at an extreme point in at most $n$ steps. □

This procedure is a special case of a more general purification algorithm described in Lewis [12]. It has been implemented on a microcomputer, and an example of the output is shown in Figure 1. The graphs show the slack variable $\zeta_r(s)$ at each iteration for the one-sided $L_1$-approximation problem

$$\text{EX1:} \quad \text{minimize} \quad \sum_{i=1}^{7} (1/i) x_i$$

$$\text{subject to} \quad \sum_{i=1}^{7} x_i s^{i-1} \geq -\sum_{i=0}^{4} s^{2i} \quad \text{for } s \in [0, 1].$$

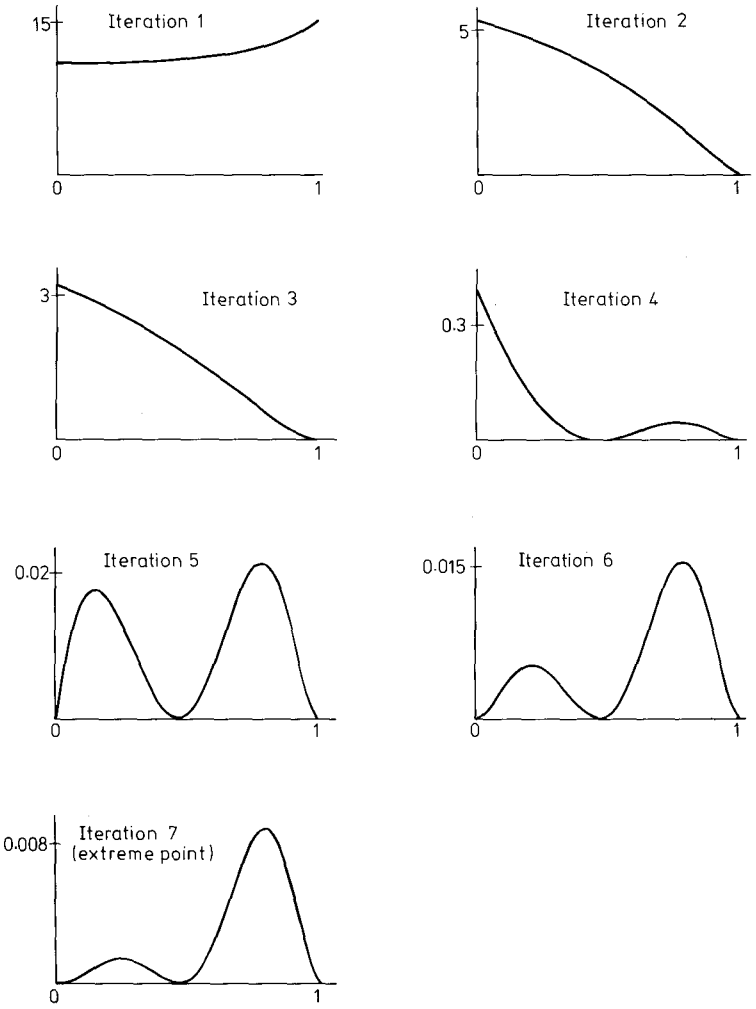


Fig. 1. Purification algorithm applied to EX1.

The starting point is $x = \xi^1 = (10, 0, 0, 0, 0, 0, 0)^T$, and the algorithm terminates after 6 steps at an extreme point, with $k = 3$, $d(1) = 2$, $d(2) = 1$, and $d(3) = 1$.

## 5. Degeneracy and the reduced cost

We consider next the problem of checking whether or not an extreme point of SIP2 is optimal, and if not, of how to make an improvement to it. Using the framework of Section 2, we shall calculate the reduced cost corresponding to a non-degenerate extreme point.

**Lemma 6.** *$(\xi; \zeta)$ is a non-degenerate extreme point of* SIP2 *if and only if $\hat{A}$ is invertible.*

**Proof.** Suppose that $(\xi; \zeta)$ is extreme, and $(x; z) \in D(\xi; \zeta)$. Then for some $(u; v) \in B(\xi; \zeta)$, we have $(x-u; z-v) \in N(A)$. Thus $v^{(j)}(s_i) = 0$, $j = 1, \ldots, d(i)$, each $i = 1, \ldots, k$, and $a(s)^{\mathrm{T}}(x-u) = z(s) - v(s)$. Hence we have

$$a^{(j)}(s_i)^{\mathrm{T}} u = a^{(j)}(s_i)^{\mathrm{T}} x - z^{(j)}(s_i), \quad j = 1, \ldots, d(i), \text{ for each } i = 1, \ldots, k. \tag{3}$$

$(\xi; \zeta)$ is non-degenerate if and only if (3) is solvable for $u$, for every $x$ and $z$, or in other words for every right hand side. This is equivalent to $\hat{A}$ being invertible. □

For any $z \in C^{\infty}[0, 1]$, we define a vector $\hat{z}$ by

$$\hat{z} = (z(s_1), \ldots, z^{(d(1))}(s_1), \ldots, z(s_k), \ldots, z^{(d(k))}(s_k))^{\mathrm{T}}.$$

Thus $\hat{A}\xi = \hat{b}$. If $(\xi; \zeta)$ is an extreme point then this relationship determines its value once $s_1, s_2, \ldots, s_k, d(1), d(2), \ldots, d(k)$ are given. With this notation we can write (3) as $\hat{A}u = \hat{A}x - \hat{z}$, so for non-degenerate $(\xi; \zeta)$ we obtain $u = x - \hat{A}^{-1}\hat{z}$, since $\hat{A}$ is invertible. Thus the projection map onto $N(A)$ is given by $P_{N(A)}(x; z) = (\hat{A}^{-1}\hat{z}; a(\cdot)^{\mathrm{T}}\hat{A}^{-1}\hat{z})$, and so the reduced cost $c^*_{(\xi;\zeta)}$ is defined by

$$\begin{aligned}\langle (x; z), c^*_{(\xi;\zeta)} \rangle &= \langle P_{N(A)}(x; z), c^* \rangle \\ &= c^{\mathrm{T}}\hat{A}^{-1}\hat{z}.\end{aligned}$$

We define $n$ scalars $\lambda_{i,j}$ by

$$\begin{aligned}c^{\mathrm{T}}\hat{A}^{-1} &= (\lambda_{1,0}, \ldots, \lambda_{1,d(1)}, \ldots, \lambda_{k,0}, \ldots, \lambda_{k,d(k)}) \\ &= \hat{\lambda}^{\mathrm{T}}.\end{aligned} \tag{4}$$

We then obtain the following optimality check:

**Theorem 7.** *A non-degenerate extreme point $(\xi; \zeta)$ is optimal if and only if for each $i = 1, \ldots, k$, $\lambda_{i,0} \geqslant 0$ and $\lambda_{i,j} = 0$, $j = 1, \ldots, d(i)$.*

**Proof.** By Lemma 2, $(\xi; \zeta)$ is optimal if and only if

$$\sum_{i=1}^{k} \sum_{j=0}^{d(i)} \lambda_{i,j} z^{(j)}(s_i) \geqslant 0$$

for all $z \geqslant 0$ on $[0, 1]$. Now suppose that $\lambda_{p,0} < 0$ for some $p$. By constructing a polynomial $z$, non-negative on $[0, 1]$ and satisfying

$$\begin{aligned}z^{(j)}(s_i) &= 1, \quad j = 0, \ i = p, \\ &= 0, \quad \text{otherwise},\end{aligned}$$

it may be seen that $(\xi; \zeta)$ cannot be optimal. If, on the other hand, $\lambda_{p,q} \neq 0$ for some $p$ and some $q > 0$ then by constructing a non-negative $z$ satisfying

$$\begin{aligned}z^{(j)}(s_i) &= 1, \quad j = 0, \ i = p, \\ &= M, \quad j = q, \ i = p, \\ &= 0, \quad \text{otherwise},\end{aligned}$$

with $|M|$ very large, it is again clear that $(\xi; \zeta)$ cannot be optimal. The result follows. □

## 6. The local structure of the feasible region, and descent steps

In this section we will consider a local description of the feasible region by a finite number of implicitly defined inequality constraints. Using this description and the optimality check described in Theorem 7, we show how a non-degenerate extreme point $(\xi; \zeta)$ may be moved to a strictly improved extreme point (if it is not already optimal).

We shall consider first the simplest case, with $(\xi; \zeta)$ not necessarily extreme, $\{s_1, \ldots, s_k\} \subset (0, 1)$, and $d(i) = 1$ for each $i = 1, \ldots, k$. The feasible region in a neighbourhood of $(\xi; \zeta)$ may then be described by $k$ inequality constraints. The following result is a special case of the 'Constraint Reduction Lemma' of Hettich and Jongen [10]. We denote the open ball $\{x: \|x - \xi\| < \delta\}$ by $N_\delta(\xi)$.

**Theorem 8.** *For some $\delta > 0$ there exist functions $w_1, \ldots, w_k \in C^\infty[N_\delta(\xi)]$ such that for all $x \in N_\delta(\xi)$, $(x; a(\cdot)^T x - b(\cdot))$ is feasible for* SIP2 *if and only if $w_i(x) \geq 0$ for each $i = 1, \ldots, k$.*

**Proof.** For $\delta$ sufficiently small and any $x \in N_\delta(\xi)$, the slack variable $a(\cdot)^T x - b(\cdot)$ has a unique local minimum close to $s_i$. We define $w_i(x)$ as the value of this local minimum. More precisely, by the Implicit Function Theorem, for a sufficiently small neighbourhood $N_\delta(\xi)$ we can define functions $\phi_i: N_\delta(\xi) \to (0, 1)$ satisfying $\phi_i(\xi) = s_i$, for each $i = 1, \ldots, k$ by

$$a'(\phi_i(x))^T x = b'(\phi_i(x)).$$

Now for $x$ sufficiently close to $\xi$, the global minimum of $a(\cdot)^T x - b(\cdot)$ on $[0, 1]$ will occur at $\phi_l(x)$ for some $1 \leq l \leq k$. This is because, for $x$ sufficiently close to $\xi$, the points $\phi_i(x)$ remain local minima of $a(\cdot)^T x - b(\cdot)$, and other local minima will have larger values. We now define functions $w_i: N_\delta(\xi) \to \mathbb{R}$ by

$$w_i(x) = a(\phi_i(x))^T x - b(\phi_i(x)) \quad \text{for each } i = 1, \ldots, k.$$

For $x$ sufficiently close to $\xi$, $a(s)^T x - b(s) \geq 0$ on $[0, 1]$ if and only if $w_i(x) \geq 0$ for each $i = 1, \ldots, k$. The result follows. □

Let us now suppose that $(\xi; \zeta)$ is a non-degenerate extreme point, still with $\{s_1, \ldots, s_k\} \subset (0, 1)$ and $d(i) = 1$, each $i = 1, \ldots, k$, and that the optimality check described in Theorem 7 fails. The optimality check is in two parts, and we consider the two cases separately.

*Case 1*: $\lambda_{j,0} < 0$ for some $j$. Denoting the $r$th unit vector in $\mathbb{R}^n$ by $e_r$, let $g = \hat{A}^{-1} e_{2j-1}$, so that $g$ has the property that $a(s_i)^T g = \delta_{ij}$, and $a'(s_i)^T g = 0$, each $i = 1, \ldots, k$. Thus

for $\varepsilon > 0$ sufficiently small we have $\phi_i(\xi + \varepsilon g) = s_i$, and $w_i(\xi + \varepsilon g) = \varepsilon \delta_{ij}$ in the notation of Theorem 8, so $\xi + \varepsilon g$ is feasible. Moreover, $c^{\mathrm{T}} g = \lambda_{j,0} < 0$, so $g$ is a descent direction. Moving as far as possible in this direction, our new point will be $\xi' = \xi + \alpha g$, where

$$1/\alpha = \beta = \sup\left\{ -\frac{a(s)^{\mathrm{T}} g}{\zeta(s)} : s \in [0, 1],\ s \neq s_1, \ldots, s_k \right\}, \tag{5}$$

just as in the purification algorithm. This step may be thought of as increasing the slack at the active point $s_j$, and is analogous to the pivot in classical linear programming. If the new point is not already extreme, we can apply the purification algorithm.

*Case 2*: $\lambda_{j,1} \neq 0$ for some $j$. In this case we consider perturbing the positions of the active points. Define $\tau \in \mathbb{R}^k$ by $\tau = (s_1, \ldots, s_k)^{\mathrm{T}}$. For $t \in \mathbb{R}^k$ define

$$\hat{A}(t) = (a(t_1), a'(t_1), \ldots, a(t_k), a'(t_k))^{\mathrm{T}}, \tag{6}$$

$$\hat{b}(t) = (b(t_1), b'(t_1), \ldots, b(t_k), b'(t_k))^{\mathrm{T}}. \tag{7}$$

Since $\hat{A}(\tau)$ is invertible by assumption, for some $\delta_1 > 0$, $\hat{A}(t)$ is invertible for $t \in N_{\delta_1}(\tau)$. For such $t$ define $x(t)$ as $(\hat{A}(t))^{-1}\hat{b}(t)$. Notice that $x(\tau) = \xi$, and for some $\delta_2 > 0$, $x(t)$ is feasible for $t \in N_{\delta_2}(\tau)$ because $\phi_i(x(t)) = t_i$, and $w_i(x(t)) = 0$, for each $i = 1, \ldots, k$. As $\hat{A}(t)x(t) = \hat{b}(t)$ we obtain

$$\frac{\partial \hat{A}}{\partial t_i} x(t) + \hat{A}(t) \frac{\partial x}{\partial t_i} = \frac{\partial \hat{b}}{\partial t_i} \quad \text{for each } i = 1, \ldots, k,$$

and so

$$\left.\frac{\partial x}{\partial t_i}\right|_{\tau} = -\hat{A}^{-1}\left(\frac{\partial \hat{A}}{\partial t_i}\xi - \frac{\partial \hat{b}}{\partial t_i}\right)\Bigg|_{\tau}.$$

Notice that the only non-zero component in $((\partial \hat{A}/\partial t_i)\xi - (\partial \hat{b}/\partial t_i))|_{\tau}$ is $a^{(2)}(s_i)^{\mathrm{T}}\xi - b^{(2)}(s_i) = \zeta^{(2)}(s_i)$, as $\zeta^{(1)}(s_i) = 0$. From this we deduce that

$$\frac{\partial}{\partial t_i}(c^{\mathrm{T}} x(t))|_{\tau} = -\lambda_{i,1}\zeta^{(2)}(s_i) \quad \text{for each } i = 1, \ldots, k.$$

We thus have the derivative of the cost with respect to movements in $t$-space (the space parametrizing the active points).

We can now use the above gradient information to perform a search in $\mathbb{R}^k$ ($t$-space). We can either choose to move all the active points simultaneously at each step, or to move only one at a time. The former option will give steeper descent steps, but the latter may be easier computationally since at each step only two rows of $\hat{A}(t)$ will change, allowing a more efficient calculation of $(\hat{A}(t))^{-1}$. This is the method which has been implemented.

Having chosen the descent direction ($h$, say) in $t$-space, we can perform a constrained line search, minimizing $c^{\mathrm{T}} x(\tau + \alpha h)$ over $\alpha \geqslant 0$. In general, as we increase $\alpha$, $x(\tau + \alpha h)$ will eventually become infeasible. If this happens before a local minimum of $c^{\mathrm{T}} x(\tau + \alpha h)$ is reached then we need to calculate the precise value of $\alpha$ for which it occurs. Either a new point becomes active, or $a^{(2)}(t_i)^{\mathrm{T}} x(\tau + \alpha h) - b(t_i)$

becomes zero for some $i$. Consider the first possibility. To find the exact $x(t)$ for which the new point becomes active, we solve:

$$a(s_{\text{new}})^{\mathrm{T}}x(\tau+\alpha h)=b(s_{\text{new}}),$$
$$a'(s_{\text{new}})^{\mathrm{T}}x(\tau+\alpha h)=b'(s_{\text{new}}),$$

(two nonlinear equations in two unknowns, $s_{\text{new}}$, the new active point, assumed to lie in $(0, 1)$, and $\alpha$, the step length) using Newton–Raphson for instance. The second possible reason for infeasibility is dealt with similarly, and is straightforward.

At this point we can summarize the steps of the algorithm as follows:

1. Find an initial feasible solution, $(\xi_0; \zeta_0)$.
2. Use the purification algorithm to find an initial extreme point, $(\xi_1; \zeta_1)$. Set $r=1$.

*Iteration r*

3. Set $\tau=(s_1, s_2, \ldots, s_k)^{\mathrm{T}}$, with coefficients the active points for $(\xi_r; \zeta_r)$. Calculate $\hat{A}$ from (1) and $\hat{\lambda}$ from (4). We assume that $\hat{A}$ is of full rank.
4. If $\lambda_{j,0}<0$ for some $j$, set $g=\hat{A}^{-1}e_{2j-1}$, and $x=\xi_r+\alpha g$, where $\alpha$ is determined from (5). Set $z(\cdot)=a(\cdot)^{\mathrm{T}}x-b(\cdot)$. Apply the purification algorithm to $(x; z)$ (if necessary) to obtain a new improved extreme point, $(\xi_{r+1}; \zeta_{r+1})$. Increase $r$ by 1. Go to 3.
5. If $\lambda_{j,1}\neq 0$ for some $j$, set $h=e_j$ and write $x(t)$ for $\hat{A}(t)^{-1}\hat{b}(t)$, where $\hat{A}(t)$ and $\hat{b}(t)$ are defined by (6) and (7). Now carry out a constrained line search to find $\bar{\alpha}$, the choice of $\alpha$ which minimizes $c^{\mathrm{T}}x(\tau+\alpha h)$ subject to $x(\tau+\alpha h)$ remaining feasible (see the remarks in the above paragraph). Set $\xi_{r+1}=x(\tau+\bar{\alpha}h)$. Increase $r$ by 1. Go to 3.

At step 1, the choice of initial feasible solution may be obvious. If not, it can be found using a phase 1 procedure which solves the semi-infinite program (posed over $\mathbb{R}^{n+1}$)

$$\begin{aligned}&\text{minimize} && x_0\\ &\text{subject to} && x_0+a(s)^{\mathrm{T}}x\geq b(s) \text{ for all } s\in[0,1],\\ &&& x_0\in\mathbb{R},\ x\in\mathbb{R}^n,\end{aligned}$$

stopping as soon as a feasible solution is reached in which $x_0\leq 0$.

Up to now we have assumed that $\{s_1, \ldots, s_k\}\subset(0, 1)$, and that $\zeta^{(2)}(s_i)>0$, for $i=1, \ldots, k$. We suppose now that this last condition does not hold, so that $d(l)>1$ for some $l$. As previously observed, $d(l)$ must be odd, so for illustration consider the case $d(l)=3$. We need to consider a variety of different descent steps. One way to keep track of changes in the objective function is to observe that, for $(x; z)$ any other feasible solution,

$$\begin{aligned}c^{\mathrm{T}}x-c^{\mathrm{T}}\xi&=c^{\mathrm{T}}(x-\xi)\\&=\hat{\lambda}^{\mathrm{T}}\hat{A}(x-\xi)\\&=\hat{\lambda}^{\mathrm{T}}(\hat{A}x-\hat{b})\\&=\hat{\lambda}^{\mathrm{T}}\hat{z},\end{aligned}$$

where $\hat{\lambda}$ is defined by (4). Hence if $x$ is obtained by some perturbation maintaining all the active points except $s_l$ unchanged then the change in the objective function is given by

$$c^{\mathrm{T}}x - c^{\mathrm{T}}\xi = \lambda_{l,0}z(s_l) + \lambda_{l,1}z'(s_l) + \lambda_{l,2}z^{(2)}(s_l) + \lambda_{l,3}z^{(3)}(s_l). \tag{8}$$

Consider the effect of splitting the active point $s_l$ into two new active points at $s_l+\delta_1$ and $s_l+\delta_2$. Thus we define $x(\delta_1, \delta_2)$ by

$$a^{(j)}(s_i)^{\mathrm{T}}x(\delta_1, \delta_2) = b^{(j)}(s_i), \quad j=0, \ldots, d(i), \quad i \neq l,$$

$$a^{(j)}(s_l+\delta_p)^{\mathrm{T}}x(\delta_1, \delta_2) = b^{(j)}(s_l+\delta_p), \quad j=0, 1, \quad p=1, 2,$$

for sufficiently small $\delta_1 \neq \delta_2$, and

$$a^{(j)}(s_i)^{\mathrm{T}}x(\delta_1, \delta_1) = b^{(j)}(s_i), \quad j=0, \ldots, d(i), \quad i \neq l,$$

$$a^{(j)}(s_l+\delta_1)^{\mathrm{T}}x(\delta_1, \delta_1) = b^{(j)}(s_l+\delta_1), \quad j=0, 1, 2, 3.$$

$x(\delta_1, \delta_2)$ is then continuous in $(\delta_1, \delta_2)$, with $x(0, 0) = \xi$. Assuming that the corresponding slack variable has a Taylor expansion for small $(s-s_l)$, $\delta_1$, $\delta_2$ of order of magnitude $O(\delta)$, we have

$$a(s)^{\mathrm{T}}x(\delta_1, \delta_2) - b(s) = K(s-s_l-\delta_1)^2(s-s_l-\delta_2)^2 + O(\delta^5),$$

for some constant $K$, since the slack has double roots at $s_l+\delta_1$, $s_l+\delta_2$. Equating coefficients of $(s-s_l)^4$ we obtain

$$\begin{aligned} K &= \tfrac{1}{24}(a^{(4)}(s_l)^{\mathrm{T}}\xi - b^{(4)}(s_l)) \\ &= \tfrac{1}{24}\zeta^{(4)}(s_l). \end{aligned}$$

Thus, from (8), we obtain that the change in the objective function when we make this perturbation is given by

$$\begin{aligned} c^{\mathrm{T}}x(\delta_1, \delta_2) - c^{\mathrm{T}}\xi = \tfrac{1}{24}\zeta^{(4)}(s_l)(&-12\lambda_{l,3}(\delta_1+\delta_2) + 2\lambda_{l,2}(\delta_1^2+4\delta_1\delta_2+\delta_2^2) \\ &- 2\lambda_{l,1}\delta_1\delta_2(\delta_1+\delta_2) + \lambda_{l,0}\delta_1^2\delta_2^2) + O(\delta^5). \end{aligned}$$

Using this formula and (8) we obtain the following as possible descent steps (without loss of generality we take $l=k$):

(a) $\lambda_{k,0} < 0$; as Case 1 above.

(b) $\lambda_{k,2} < 0$; let $g = \hat{A}^{-1}e_{n-1}$. Then $\xi + \varepsilon g$ is an improved solution, for small $\varepsilon > 0$. This move increases $\zeta^{(2)}(s_k)$, and as in Case 1 above, we need to move as far as possible in this direction and then possibly purify to obtain a new extreme point.

(c) $\lambda_{k,3} \neq 0$; move $s_k$, keeping $d(k) = 3$.

(d) $\lambda_{k,3} = 0$ and $\lambda_{k,2} > 0$; replace $\{s_1, s_2, \ldots, s_k\}$ with $\{s_1, \ldots, s_{k-1}, s_k - \delta, s_k + \delta\}$, and take $d(k) = d(k+1) = 1$.

(e) $\lambda_{k,3} = \lambda_{k,2} = 0$ and $\lambda_{k,1} \neq 0$; move $s_k$, keeping $d(k) = 3$.

The other situation which we need to consider is when one of the active points is 0 or 1. Suppose for example that $s_1 = 0$ and $d(1) = 1$. The only case which causes

difficulty is when $\lambda_{1,1} < 0$: Case 2 above indicates that we should decrease $s_1$, which is not possible. We can however move by increasing the derivative of the slack at 0. Define $g$ by

$$a^{(j)}(s_i)^T g = 0, \quad j = 0, \ldots, d(i), \quad i = 2, \ldots, k,$$

$$a(s_1)^T g = 0,$$

$$a'(s_1)^T g = 1.$$

Then for $\varepsilon > 0$ sufficiently small, $\xi + \varepsilon g$ is feasible, and since $c^T g = \lambda_{1,1} < 0$, $g$ is a descent direction.

Thus we have shown that whenever the optimality check fails an improved extreme point can be found, using one of the methods outlined above. Hence we have derived a descent method for the primal semi-infinite problem analogous to the simplex algorithm. We have no general result guaranteeing that the method will converge to an optimal solution, but the descent steps described above have beeen implemented in an algorithm to solve SIP2 on a microcomputer, and in practice the method works well, in the absence of degeneracy. The question of local convergence is considered in the following section. We illustrate this by describing the performance of the algorithm for two small examples.

First consider the problem EX1 introduced in Section 4. The non-degenerate extreme point found by the purification algorithm (see Section 4) is used as an initial point. Figure 2 shows graphs of the slack variable at various stages of the solution procedure. Notice that during the course of the calculation the active point at 0 is split into two new active points, one at 0 and one in (0, 1). The algorithm terminates at the optimum (to a given tolerance).

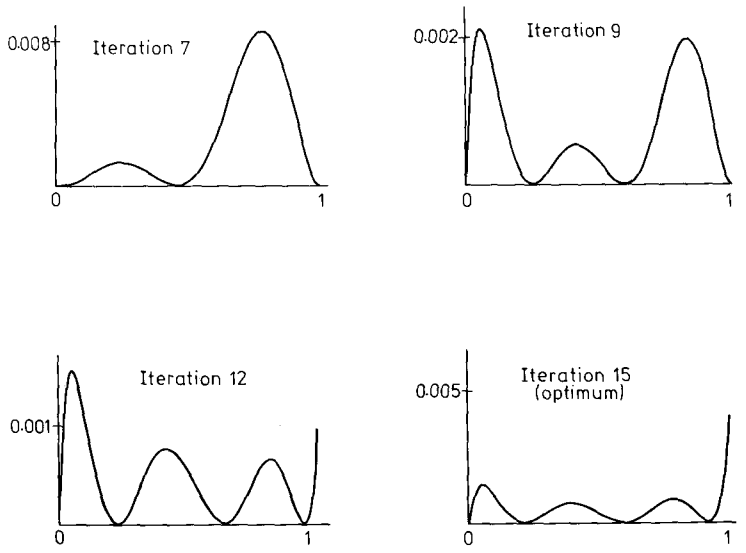


Fig. 2. Changes in slack variable for the algorithm applied to EX1.

Our second example is the following well known test problem (due to Roleff [15]):

$$\text{minimize} \quad \sum_{i=1}^{n} (1/i)x_i$$

$$\text{subject to} \quad \sum_{i=1}^{n} x_i s^{i-1} \geq \tan(s) \text{ for all } s \in [0, 1].$$

This problem arises from the one-sided $L_1$-approximation of $\tan(s)$ on $[0, 1]$ by polynomials of degree less than $n$. Coope and Watson [3] observe that it is extremely ill-conditioned for $n > 6$. The problem was solved for various values of $n$ by the new algorithm. The results are shown below.

$n = 3$: initial $x = (2, 0, 0)^T$; 4 iterations (2 purification steps, 2 further descent steps); optimal value = 0.649042; optimal $x = (0.089232, 0.422510, 1.045665)^T$; active points $\{0.333, 1\}$.

$n = 6$: initial $x = (2, 0, 0, 0, 0, 0)^T$: 12 iterations (6 purification steps, 6 further descent steps); optimal value = 0.61608515; optimal $x = (0, 1.023223, -0.240305, 1.220849, -1.387306, 0.940948)^T$; active points $\{0, 0.276, 0.723, 1\}$.

$n = 9$: the cases $n = 6$, 7, 8 and 9 were solved sequentially, each time using the previous solution as the initial $x$ for the next problem. The cases $n = 7$, 8 and 9 took respectively 8, 6 and 10 iterations. For $n = 9$, optimal value = 0.61563261; optimal $x = (0.000033, 0.998329, 0.029955, 0.089219, 1.055433, -2.459376, 3.653543, -2.728758, 0.919029)^T$; active points $\{0.055, 0.276, 0.582, 0.860, 1\}$.

In both of the above examples the algorithm was terminated when an extreme point was reached for which the reduced cost coefficients satisfied $\lambda_{i,0} > -10^{-3}$ and $|\lambda_{i,j}| < 10^{-3}$, $j = 1, \ldots, d(i)$, each $i = 1, \ldots, k$. It happens that in these examples all the descent steps, after finding an initial extreme point, are of the second type. These steps are performed by moving only one active point at a time: the reduced cost is recalculated at each iteration, allowing the search to be performed by bisection. More accurate results could be obtained by reducing the tolerance in the termination criterion, at the expense of increasing the number of iterations required.

One of the main practical difficulties with the new algorithm is that we often have to check that a new $(\xi; \zeta)$ is feasible for the problem. For example, this occurs frequently during the line search in step 5 of the algorithm. In order to do this we need to find all the local minima of the slack variable $\zeta$. Naturally, any algorithm for the solution of SILP will need to include a subroutine to accomplish this. In our implementation the local minima are simply recalculated at each step, using a grid search followed by Newton-Raphson. The same technique is used in the calculation of $\beta_r$ in step 4 of the purification algorithm, and in the calculation of $\alpha$ in (5).

There is clearly some scope for refinement in the numerical implementation of these local minima computations. For example in the intial stages of the line search we could afford to compute these minima less accurately, while in the later stages we could use the local minima of $a(\cdot)^T\xi - b(\cdot)$ as first approximations to the corresponding local minima of $a(\cdot)^T\xi' - b(\cdot)$, for $\xi'$ close to $\xi$.

## 7. The degenerate case

In this section we shall analyse the notion of degeneracy and consider the problem of constructing a descent step from a degenerate extreme point. As will be seen, degeneracy corresponds roughly to too many points being active, and in general is likely to be a common phenomenon in this problem. Nevertheless there are classes of problem for which we can be sure that it does not occur. Consider for instance the problem

$$\begin{aligned} &\text{minimize} && c^{\mathrm{T}}x \\ &\text{subject to} && \sum_{i=1}^{n} x_i s^{i-1} \geqslant b(s) \quad \text{for all } s \in [0,1], \\ & && x \in \mathbb{R}, \end{aligned}$$

where $b(\cdot)$ has the property that $b^{(n)}(\cdot)$ has no roots in $[0, 1]$. It follows by repeated application of Rolle's theorem that any feasible slack can have at most $n$ roots in $[0, 1]$ (counted by multiplicity), and so any extreme point will be non-degenerate.

In finite linear programming, degenerate extreme points are dealt with by performing a sequence of degenerate pivots. One way of thinking of this procedure is that the problem is perturbed slightly and a sequence of small descent steps are made before a genuine descent direction is found. In the primal semi-infinite linear program such a perturbation approach will not necessarily succeed in resolving the degeneracy, because degenerate extreme points group together in manifolds on the boundary of the feasible region. This is expressed in the result below. We again consider feasible $(\xi; \zeta)$ for SIP2, with active points $\{s_1, \ldots, s_k\} \subset (0, 1)$, and $\zeta^{(2)}(s_i) > 0$ for each $i = 1, \ldots, k$. We consider subsets $I$ of $\{1, \ldots, k\}$, and we make the following regularity assumptions:

(a) $\{a(s_i): i \in I\}$ is linearly independent for any $I$ *with* $|I| \leqslant n$, and

(b) $\{a(s_i), a'(s_i): i \in I\}$ spans $\mathbb{R}^n$ for any $I$ with $2|I| \geqslant n$.

Now, using the notation of Theorem 8, let us partition the feasible points in a small neighbourhood $N_\delta(\xi)$ into subsets in the following fashion. For any $I \subseteq \{1, \ldots, k\}$ define $E_I \subseteq N_\delta(\xi)$ by

$$\begin{aligned} x \in E_I \quad \text{if and only if} \quad & w_i(x) = 0, \quad i \in I, \\ & > 0, \quad i \notin I. \end{aligned}$$

**Theorem 9.** *There exists a* $\delta > 0$ *such that for all* $x \in N_\delta(\xi)$, $(x; a(\cdot)^{\mathrm{T}}x - b(\cdot))$ *is an extreme point of* SIP2 *if and only if* $x \in E_I$ *for some* $I$ *with* $2|I| \geqslant n$. *Also, each* $E_I$ *is a manifold of dimension* $\max\{0, n - |I|\}$.

**Proof.** Observe that $E_I$ is just the set of feasible $x$ close to $\xi$ with active points near $\{s_i: i \in I\}$. The result follows from Theorems 8 and 4. Since

$$w_i(x) = a(\phi_i(x))^{\mathrm{T}}x - b(\phi_i(x)),$$

it follows that

$$\nabla w_i(x) = \nabla \phi_i(x)(a'(\phi_i(x))^{\mathrm{T}} x - b'(\phi_i(x))) + a(\phi_i(x)),$$

so that $\nabla w_i(\xi) = a(s_i)$ (using $\phi_i(\xi) = s_i$). The assertion concerning the dimension of $E_I$ follows, since the dimension of $\operatorname{span}\{\nabla w_i(\xi): i \in I\}$ is therefore $|I|$. □

To illustrate the sets $E_I$, consider the problem

$$\begin{aligned} &\text{minimize} && x_1 \\ &\text{subject to} && x_1 + x_2 s + x_3 s^2 \geq b(s) \text{ for all } s \in [0, 1]. \end{aligned}$$

This problem consists of minimizing the intercept at 0 of parabolas lying over the curve $y = b(s)$. In Figure 3, three feasible parabolas are illustrated, corresponding to three feasible points $(\xi^1; \zeta_1)$, $(\xi^2; \zeta_2)$, and $(\xi^3; \zeta_3)$. In $x$-space the feasible region is the convex hull of three curved lines of degenerate extreme points, emanating

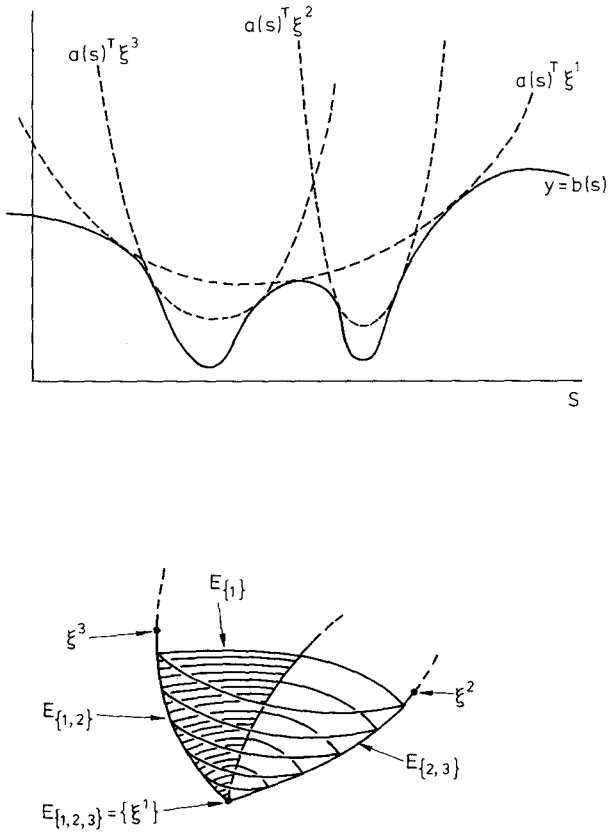


Fig. 3. Three feasible parabolas and the feasible region.

from the degenerate extreme point $\xi^1$ (see Figure 3). A detailed investigation of the local structure of the feasible region may be found in Jongen and Zwier [11].

The simplest case is when $(\xi; \zeta)$ is a non-degenerate extreme point with active points $\{s_1, \ldots, s_{n/2}\} \subset (0, 1)$, and each $d(i) = 1$. In this case Theorem 9 shows that the extreme points in a neighbourhood of $(\xi; \zeta)$ are those $(x; a(\cdot)^T x - b(\cdot))$ with $x$ lying in the manifold $E_{\{1, \ldots, n/2\}}$. Thus any extreme point sufficiently close to $(\xi; \zeta)$ will have active points $\{t_1, \ldots, t_{n/2}\}$, where $t_i$ is close to $s_i$ for each $i$. It follows that if the optimal solution is of this form then the new algorithm will be locally convergent since it reduces to an unconstrained, coordinate-wise search for a local minimum in $t$-space (the space parametrizing the active points). Of course, convergence could be improved by using a more sophisticated search strategy such as a Newton method when we are sufficiently close to the optimum. Such two-phase approaches are well-known in semi-infinite programming (see for instance Hettich [8] and [9]).

The essence of the descent step in Case 2 of Section 6 is that we can move the active points around independently whilst retaining feasibility. Degeneracy causes two difficulties. Firstly the reduced cost is no longer defined on the whole of $X = \mathbb{R}^n \times C^\infty[0, 1]$, and so no longer provides a simple optimality check, and secondly we can no longer move the active points independently. Degeneracy may be roughly thought of as too many points becoming active: ensuring feasibility by fixing the value of the slack and its derivative at all active points is no longer possible as it was in the non-degenerate case.

Let us look again at the local structure of the feasible region, but instead of using the Implicit Function Theorem to infer the existence of the unknown functions $w_i$, let us work in a larger space of points $x$ together with associated active points specified by $(t_1, \ldots, t_k)$, a point in $\mathbb{R}^k$. Define $F \subseteq \mathbb{R}^{n+k}$ by

$$F = \{(x; t): a(t_i)^T x \geqslant b(t_i),\ a'(t_i)^T x = b'(t_i),\ i = 1, \ldots, k\}.$$

**Theorem 10.** *Suppose* $(\xi; \zeta)$ *is feasible for* SIP2, *with active points* $\{s_1, \ldots, s_k\} \subset (0, 1)$, *and* $\zeta^{(2)}(s_i) > 0$ *for each* $i = 1, \ldots, k$. *Define* $\tau \in \mathbb{R}^k$ *by* $\tau = (s_1, \ldots, s_k)$. *Then there is a* $\delta > 0$ *such that for all* $(x; t) \in N_\delta(\xi; \tau)$, $(x; a(\cdot)^T x - b(\cdot))$ *is feasible for* SIP2 *if and only if* $(x; t) \in F$.

**Proof.** This is essentially a restatement of Theorem 8. □

Still treating $\xi$ as a fixed point, we now consider the finite problem:

RP: minimize $c^T x$

subject to $(x; t) \in F$.

By Theorem 10, $(\xi; \tau)$ is a local minimum for RP if and only if $(\xi; \zeta)$ is a local minimum and hence optimal for SIP2. The tangent space to $F$ at $(\xi; \tau)$, which we shall denote $M$, is given by

$$M = \{(x; t): a(s_i)^T x = 0 \text{ and } a'(s_i)^T x + \zeta^{(2)}(s_i) t_i = 0,\ i = 1, \ldots, k\}.$$

We can make a descent step by moving a small distance in the direction $-P_M(c; 0)$ (where $P_M$ is the orthogonal projection onto $M$), followed by a restoration step to return us to the feasible region, $F$. These will be accomplished using standard techniques from the projected gradient algorithm (see for instance Gill, Murray and Wright [4]). If $P_M(c; 0) = 0$ then we have

$$\sum_{i=1}^{k} \mu_i a(s_i) + \sum_{i=1}^{k} \nu_i a'(s_i) = c \quad \text{and} \quad \nu_j \zeta^{(2)}(s_j) = 0 \text{ for } j = 1, \ldots, k,$$

for some $\mu, \nu \in \mathbb{R}^k$. Since $\zeta^{(2)}(s_j) > 0$ by assumption, we have $\sum_{i=1}^{k} \mu_i a(s_i) = c$. If $\mu_i \geq 0$, for $i = 1, \ldots, k$, then $(\xi; \tau)$ satisfies the first order (Kuhn–Tucker) optimality conditions for RP, and the projected gradient algorithm terminates. Interpreted as a measure on the points $s_1, \ldots, s_k$, $\mu$ is in this case a feasible solution to the dual problem for SIP2,

$$\begin{aligned} \text{SIP2}^*\text{: maximize} \quad & \int_0^1 b(s)\, \mathrm{d}\omega(s) \\ \text{subject to} \quad & \int_0^1 a(s)\, \mathrm{d}\omega(s) = c, \\ & \omega \geq 0, \ \omega \in M[0, 1], \end{aligned}$$

and is complementary slack with $\xi$, so that both $\xi$ and $\mu$ are optimal for their respective programs (see Nash [13]). Suppose $\mu$ is not non-negative. The standard projected gradient algorithm would then drop the constraint corresponding to the most negative component of $\mu$, $\mu_j$ say, and move in the direction of the negative cost vector projected onto the subspace determined by the remaining active constraints. In this case, dropping the constraint associated with $\mu_j$ means increasing the value of the slack at $t_j$. Thus we are no longer interested in the precise value of $t_j$ and we can simplify calculations by working in the smaller set

$$F' = \{(x; t): a(t_i)^{\mathrm{T}} x \geq b(t_i) \text{ and } a'(t_i)^{\mathrm{T}} x = b'(t_i),\ i \neq j\}.$$

As a final point, notice that the treatment we gave of the non-degenerate case made use of the fact that we can write

$$\begin{aligned} &\{(x; t): a(t_i)^{\mathrm{T}} x = b(t_i),\ a'(t_i)^{\mathrm{T}} x = b'(t_i),\ i = 1, \ldots, k\} \\ &\quad = \{(\hat{A}(t)^{-1} \hat{b}(t); t): t \in \mathbb{R}^k\}, \end{aligned}$$

so that moving in the set $F$ is in this case straightforward. It remains to be seen whether the special structure of $F$ allows an analogous simplification of calculations in the degenerate case.

## 8. Higher dimensions

We finally return to the problem SIP1 when $S$ is a polyhedral subset of $\mathbb{R}^p$. We shall suppose that $a_1, \ldots, a_n, b \in C^2[S]$. With the addition of a slack variable, the problem

becomes

$$\begin{aligned} \text{SIP3:} \quad & \text{minimize} \quad c^{\mathrm{T}}x \\ & \text{subject to} \quad a(s)^{\mathrm{T}}x - z(s) = b(s) \quad \text{for all } s \in S, \\ & \qquad x \in \mathbb{R}^n, \ z \in C^2[S], \ z \geqslant 0, \end{aligned}$$

where $S \subseteq \mathbb{R}^p$ is a compact set defined by

$$S = \{s: d_j^{\mathrm{T}} s \leqslant f_j, \ j = 1, \ldots, q\}.$$

We consider a $(\xi; \zeta)$ feasible for SIP3, and denote the active points $\{s \in S: \zeta(s) = 0\}$ by $\{s^1, \ldots, s^k\}$ (assumed to be a finite set). For each $i = 1, \ldots, k$, let $J(i)$ be the set of indices of those constraints on $S$ which are active at $s^i$, i.e. $J(i) = \{j: d_j^{\mathrm{T}} s^i = f_j\}$. We write $\zeta_s$ for the vector $(\partial\zeta/\partial s_1, \ldots, \partial\zeta/\partial s_p)$, $\zeta_{ss}$ for the corresponding Hessian matrix, and $a_s$ for the matrix $(\partial a/\partial s_1, \ldots, \partial a/\partial s_p)$.

We shall consider only the case where $\zeta$ satisfies the second order sufficient conditions for a local minimum at each $s^i$, with strict complementary slackness:

$$\begin{aligned} & \text{For } i = 1, \ldots, k, \text{ there is a } \mu^i \in \mathbb{R}^q \text{ such that} \\ & \mu_j^i > 0, \quad j \in J(i), \\ & \quad = 0, \quad j \notin J(i), \\ & \zeta_s(s^i)^{\mathrm{T}} + \sum_{j \in J(i)} \mu_j^i d_j = 0, \quad \text{and} \\ & \zeta_{ss}(s^i) \text{ is positive definite on } \{s: d_j^{\mathrm{T}} s = 0, \ j \in J(i)\}. \end{aligned} \tag{9}$$

In the one-dimensional case where $S = [0, 1]$ this corresponds to the situation when $d(i) = 1$ for $s \in (0, 1)$ and $d(i) = 0$ for $s_i = 0$ or 1.

Now for each $i = 1, \ldots, k$, define $m(i)$ to be the dimension of the subspace $\{s: d_j^{\mathrm{T}} s = 0, \ j \in J(i)\}$. By (9) we can choose a basis $\{g_1^i, \ldots, g_{m(i)}^i\}$ for this space, satisfying $(g_j^i)^{\mathrm{T}} \zeta_{ss}(s^i) g_l^i = \delta_{jl}$. Write $G_i$ for the matrix $(g_1^i, \ldots, g_{m(i)}^i)$. We then obtain the following analogue of Lemma 3:

**Lemma 11.** *For* $(x; z) \in \mathbb{R}^n \times C^2[S]$, $(x; z) \in B(\xi; \zeta)$ *if and only if*

$$z(s^i) = 0 \ \textit{and} \ z_s(s^i) G_i = 0 \quad \textit{for each } i = 1, \ldots, k. \tag{10}$$

**Proof.** If $(x; z) \in B(\xi; \zeta)$ then $\zeta(s^i) \pm \lambda z(s^i) \geqslant 0$ for some $\lambda > 0$, so $z(s^i) = 0$ and $\zeta + \lambda z$ must satisfy the first order conditions for a local minimum at each $s^i$, $i = 1, \ldots, k$. Thus for some $\gamma^i \in \mathbb{R}^q$,

$$\zeta_s(s^i)^{\mathrm{T}} + \lambda z_s(s^i)^{\mathrm{T}} + \sum_{j \in J(i)} \gamma_j^i d_j = 0.$$

So from (9), for $l = 1, 2, \ldots, m(i)$,

$$\begin{aligned} \lambda z_s(s^i) g_l^i &= \sum_{j \in J(i)} (\mu_j^i - \gamma_j^i) d_j^{\mathrm{T}} g_l^i \\ &= 0. \end{aligned}$$

Hence $z_s(s^i) G_i = 0$, for each $i = 1, \ldots, k$, so (10) is satisfied.

Conversely, suppose $(x; z)$ satisfies (10). Then $z_s(s^i)^T$ is perpendicular to the space spanned by $\{g^i_1, \ldots, g^i_{m(i)}\}$, and so $z_s(s^i)^T = \sum_{j \in J(i)} \gamma^i_j d_j$ for some $\gamma^i_j$. Hence

$$\zeta_s(s^i)^T \pm \lambda z_s(s^i)^T + \sum_{j \in J(i)} (\mu^i_j \mp \gamma^i_j) d_j = 0,$$

and it is not hard to check that $\zeta \pm \lambda z$ satisfies the second order sufficient conditions for a local minimum at each $s^i$, for $\lambda$ sufficiently small. So $\zeta(s) \pm \lambda z(s) \geq 0$ for $s \in S$, for $\lambda$ sufficiently small. □

We now define $\hat{A}$ and $\hat{z}$ in an analogous fashion to the one-dimensional case by:

$$\hat{A} = (a(s^1), a_s(s^1)G_1, \ldots, a(s^k), a_s(s^k)G_k)^T,$$

$$\hat{z} = (z(s^1), z_s(s^1)G_1, \ldots, z(s^k), z_s(s^k)G_k)^T.$$

The analogue of Theorem 4 is then:

**Theorem 12.** *$(\xi; \zeta)$ is an extreme point of* SIP3 *if and only if the columns of $\hat{A}$ are linearly independent.*

**Proof.** $(x; z) \in B(\xi; \zeta) \cap N(A)$ if and only if $a(s^i)^T x = 0$, and $G_i^T a_s(s^i)^T x = 0$, each $i = 1, \ldots, k$, i.e. if and only if $\hat{A}x = 0$, whence the result. □

The purification algorithm described in Section 4 will operate in exactly the same fashion, if we take

$$T_r = \{x: a(s^i)^T x = 0, G_i^T a_s(s^i)^T x = 0 \text{ for each } i = 1, \ldots, k\},$$

providing that the slack $\zeta_r$ satisfies (9) at each step.

Just as in the one-dimensional case, we find that an extreme point $(\xi; \zeta)$ is non-degenerate exactly when $\hat{A}$ is invertible, and in this case the associated reduced cost is defined by

$$\langle (x; z), c^*_{(\xi;\zeta)} \rangle = c^T \hat{A}^{-1} \hat{z}.$$

Write $c^T \hat{A}^{-1} = (\lambda_{1,0}, \lambda_{1,1}, \ldots, \lambda_{1,m(1)}, \ldots, \lambda_{k,0}, \ldots, \lambda_{k,m(k)})$. Then an analogous argument to the one-dimensional case shows that $(\xi; \zeta)$ is optimal if and only if for each $i = 1, \ldots, k$, $\lambda_{i,0} \geq 0$ and $\lambda_{i,j} = 0$, for $j = 1, \ldots, m(i)$.

Condition (9) allows us to describe the feasible region in a neighbourhood of $\xi$ by $k$ inequality constraints, exactly as in the one-dimensional case (see Hettich and Jongen [10]). If $\lambda_{i,0} < 0$ for some $i$ then we can make a descent step by increasing the value of the slack at $s^i$. If on the other hand $\lambda_{i,j} \neq 0$ for some $i$ and $j > 0$, then we can make a descent step by moving $s^i$ in the direction $\pm g^i_j$. Consider for example the effect of moving the active point $s^1$ to $t \in \mathbb{R}^p$. Define

$$\hat{A}(t) = (a(t), a_s(t)G_1, a(s^2), \ldots, a_s(s^k)G_k)^T,$$

$$\hat{b}(t) = (b(t), b_s(t)G_1, b(s^2), \ldots, b_s(s^k)G_k)^T,$$

and $\hat{z}$ similarly. Define $x(t)$ as $\hat{A}(t)^{-1}\hat{b}(t)$, which is well defined for $t$ sufficiently close to $s^1$, and let the corresponding cost be $c(t) = c^{\mathrm{T}}x(t)$. Since $\hat{A}(t)x(t) = b(t)$, we have

$$x_t(s^1) = \hat{A}(s^1)^{-1}(\hat{b}_t(s^1) - \hat{A}_t(s^1)x(s^1)),$$

and so the rate of change of cost is given by

$$\begin{aligned} c_t(s^1) &= c^{\mathrm{T}}\hat{A}^{-1}(\hat{b}_t(s^1) - \hat{A}_t(s^1)\xi) \\ &= (\lambda_{1,0}, \ldots, \lambda_{k,m(k)}) \begin{pmatrix} b_s(s^1) - a_s(s^1)^{\mathrm{T}}\xi \\ G_1^{\mathrm{T}}[b_{ss}(s^1) - a_{ss}(s^1)^{\mathrm{T}}\xi] \\ 0 \\ \vdots \\ 0 \end{pmatrix} \\ &= -(\lambda_{1,0}, \ldots, \lambda_{k,m(k)})(\zeta_s(s^1), \zeta_{ss}(s^1)G_1, 0, \ldots, 0)^{\mathrm{T}}. \end{aligned}$$

From the definition of the $g_j^1$, we therefore have that $c_t(s^1)g_j^1 = -\lambda_{1,j}$, so for $\lambda_{1,j} \neq 0$ we can make a descent step by moving $s^1$ in the direction $\pm g_j^1$. Notice that this will not violate any of the active constraints on $s^1$ since $d_j^{\mathrm{T}}g_j^1 = 0$ for $j \in J(1)$, by definition.

**An example.** Consider the problem

$$\begin{aligned} &\text{minimize} && x_3 \\ &\text{subject to} && x_1 s_1 + x_2 s_2 + x_3 \geqslant -\tfrac{1}{6}[(s_1 - 1)^2 + s_2][s_1 + (2 - s_2)] \\ &&& \text{for all } 0 \leqslant s_1, s_2 \leqslant 2. \end{aligned}$$

In the above notation we have: $c = (0, 0, 1)^{\mathrm{T}}$, $a(s) = (s_1, s_2, 1)^{\mathrm{T}}$, $d_1 = \binom{-1}{0}$, $d_2 = \binom{0}{-1}$, $d_3 = \binom{1}{0}$, $d_4 = \binom{0}{1}$, $f_1 = 0$, $f_2 = 0$, $f_3 = 2$, $f_4 = 2$. We consider the point $\xi = (0, 0, 0)^{\mathrm{T}}$, with cost 0 and slack variable given by

$$\zeta(s_1, s_2) = \tfrac{1}{6}[(s_1 - 1)^2 + s_2][s_1 + (2 - s_2)].$$

We obtain the active points $s^1 = \binom{1}{0}$ and $s^2 = \binom{0}{2}$, with $J(1) = \{2\}$ and $J(2) = \{1, 4\}$. It may be checked that $\zeta$ satisfies (9) at $s^1$ and $s^2$, and that $\zeta_{ss}(s^1) = \frac{1}{6}\begin{pmatrix} 6 & 1 \\ 1 & -2 \end{pmatrix}$. We require $\{g_1^1\}$ to be a basis for $\{s: d_2^{\mathrm{T}}s = 0\}$, satisfying $(g_1^1)^{\mathrm{T}}\zeta_{ss}(s^1)g_1^1 = 1$, so we take $g_1^1 = \binom{1}{0}$, and $G_1 = \binom{1}{0}$. $G_2$ is null.

Now $\hat{A} = (a(s_1), a_s(s^1)G_1, a(s^2))^{\mathrm{T}}$, so

$$\hat{A} = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 2 & 1 \end{pmatrix}.$$

$\hat{A}$ is invertible, so $(\xi; \zeta)$ is a non-degenerate extreme point, with a reduced cost given by $\lambda_{1,0} = 1$, $\lambda_{1,1} = -1$, $\lambda_{2,0} = 0$. We make a descent step by moving $s^1$ in the direction $-g_1^1 = -\binom{1}{0}$. We find that to minimize the cost in this search direction, we wish to move $s^1$ to $\binom{0}{0}$. This corresponds to $\xi = \frac{1}{6}(3, 1, -2)$, which is easily seen to be optimal with cost $-\frac{1}{3}$.

## 9. Discussion

The usual methods for the solution of semi-infinite programming problems are essentially dual methods. Approximate solutions are generated which are infeasible for the primal problem. In fact an approximate problem is solved where the index set of the constraints, $S$, is replaced by a finite subset $\{s_1, \ldots, s_N\}$. This set is either a grid approximation to the original set $S$ or it is a subset of size $n$ which is updated iteratively using an exchange (perhaps multiple exchange) method. In this latter case the algorithm can be thought of as the simplex algorithm applied to the dual problem. However, in either case this first phase of the algorithm must be followed by a second phase in which a more or less exact solution is found using Newton's method (say) to solve a set of nonlinear equations which take into account the kind of derivative information which is at the heart of the primal algorithm described here (see for instance Hettich [9]).

A number of difficulties are associated with these standard methods. Firstly, since they are dual methods, if the algorithm is terminated before the optimum is reached then the resulting solution will not be feasible, which may be a disadvantage. Secondly, if the set $S$ is replaced by a grid approximation then the resulting finite linear program must be solved by special variants of the simplex algorithm if numerical instability is to be avoided, and convergence of the solution as the grid is refined may be slow. Similar difficulties occur with exchange methods (see Hettich [8] and [9]). Thirdly, finding an initial approximation to the solution for phase 2 of the algorithm requires the clustering of points in the output of phase 1 into a reduced set of points (approximations to the active points at the optimum, see Glashoff and Gustafson [5]). Unfortunately the clustering procedure may present difficulties, as has been observed in, for instance, Watson [17]. Finally, if the initial approximation for phase 2 is insufficiently accurate then the algorithm may not converge, and we will have to return to phase 1. For further details, see Gustafson and Kortanek [6]. These difficulties are well-known and have led to the adoption of various globally convergent methods (see for example Watson [17], Coope and Watson [3]).

We have not attempted any sophistication in the choice of a descent direction for improvement steps of the second kind. As we observed previously, if the optimal solution is non-degenerate then the algorithm reduces to an unconstrained search sufficiently close to this optimum, and so by employing a suitable search strategy such as a Newton method, we can ensure superlinear convergence. If the optimal solution is degenerate on the other hand, a standard second phase technique may be necessary to give rapid convergence. In either case the new method has the advantages that it maintains feasibility and avoids the clustering difficulties which constitute one of the main disadvantages associated with standard two phase techniques.

The main obstacle to the practical use of the new method is in the degenerate case. It is clear, for example, that the method would be unsuitable for standard

Chebychev approximation problems, where the optimal solution can be seen to be always degenerate: the well-known exchange method for this problem is extremely effective (and in fact may be viewed as working with a sequence of non-degenerate *dual* extreme points). The work we have done on degeneracy is important from a theoretical point of view, and gives a better understanding of the nature of this important phenomenon, but it still leaves considerable implementation difficulties. Our own small-scale implementation does not deal effectively with degeneracy. One of the principal attractions of the new method is that feasibility is maintained. However, as in the projected gradient algorithm, it is hard to see how to achieve this simply in the degenerate case without the computationally unattractive projection and restoration steps described in Section 7. Until this difficulty is resolved the method remains primarily of theoretical interest.

## References

[1] E.J. Anderson, "A new primal algorithm for semi-infinite linear programming," in: [2] pp. 108–122.
[2] E.J. Anderson and A.B. Philpott, *Infinite Programming, Proceedings* (Springer-Verlag, Berlin, 1985).
[3] I.D. Coope and G.A. Watson, "A projected lagrangian algorithm for semi-infinite programming," *Mathematical Programming* 32 (1985) 337–356.
[4] P.E. Gill, W. Murray and M. Wright, *Practical Optimization* (Academic Press, London, 1981).
[5] K. Glashoff, and S.-A. Gustafson, *Linear Optimization and Approximation* (Springer-Verlag, New York, 1983).
[6] S.-A. Gustafson and K.O. Kortanek, "Numerical treatment of a class of semi-infinite programming problems," *Naval Research Logistics Quarterly* 20 (1973) pp. 477–504.
[7] R. Hettich, *Semi-Infinite Programming, Proceedings of a Workshop* (Springer-Verlag, Berlin, 1979).
[8] R. Hettich, "A comparison of some numerical methods for semi-infinite programming," in: [7] pp. 112–125.
[9] R. Hettich, "A review of numerical methods for semi-infinite optimization," in: A.V. Fiacco and K.O. Kortanek, eds., *Semi-Infinite Programming and Applications, Proceedings of an international symposium* (Springer-Verlag, Berlin, 1983) pp. 158–178.
[10] R. Hettich, and H.Th. Jongen, "Semi-infinite programming: conditions of optimality and applications," in: J. Stoer, ed., *Optimization Techniques, Part 2* (Springer-Verlag, Berlin, 1978) pp. 1–11.
[11] H.Th. Jongen and G. Zwier, "On the local structure of the feasible set in semi-infinite optimization," in: B. Brosowski and F. Deutsch, eds., *Parametric Optimization and Approximations* (Birkhauser Verlag, Basel, 1985) pp. 185–202.
[12] A.S. Lewis, "Extreme points and purification algorithms in general linear programming," in: [2] pp. 123–135.
[13] P. Nash, "Algebraic fundamentals of linear programming," in: [2] pp. 37–52.
[14] A.F. Perold, "Extreme points and basic feasible solutions in continuous time linear programming," *SIAM Journal on Control and Optimization* 19 (1981) pp. 52–63.
[15] K. Roleff, "A stable multiple exchange algorithm for linear SIP," in: [7] pp. 83–96.
[16] W. Rudin, *Real and Complex Analaysis* (McGraw-Hill, New York, 1966).
[17] G.A. Watson, "Lagrangian methods for semi-infinite programming problems," in: [2] pp. 90–107.