ZASTOSOWANIA MATEMATYKI
APPLICATIONES MATHEMATICAE
20,2 (1990), pp. 307–327

A. KORYTOWSKI (Kraków)

INNER SEARCH METHODS FOR LINEAR PROGRAMMING

A class of linear programming algorithms is proposed. They differ from simplex and reduced gradient methods in a systematic use of inner (strictly admissible) search directions. The purpose is to implement the idea that one step along a search direction pointing to the relative interior of the admissible set can be more effective than many steps along its relative boundary. Finiteness of the algorithms is proved and some computational aspects are discussed.

1. Introduction. Consider the following linear programming problem, called LP. Minimize the cost function $x \mapsto A_0 x$ subject to $x \in W \subset R^n$, where the admissible set W is a closed and convex polyhedron determined by

$$A_i x \leqslant b_i,\ i \in C_I, \qquad A_i x = b_i,\ i \in C_E, \tag{1.1}$$
$$C_I \cup C_E = \{1, \ldots, m\}, \qquad C_I \cap C_E = \emptyset, \qquad C_I \neq \emptyset,$$
$$A_i \in R_n, \qquad A_i \neq 0, \qquad i = 0, 1, \ldots, m.$$

R^n (resp. R_n) is the Euclidean space of n-dimensional column (resp. row) vectors. The inner search methods that we wish to present are a class of methods of admissible directions and share their general properties (for a brief description see [6]). They can be decomposed into two levels. The upper level algorithm formulates a sequence of problems LLP(k), $k = 1, 2, \ldots$, which are solved by the lower level algorithm. LLP(k) has the following form:

Minimize $x \mapsto a(k)x$ subject to $x \in w(k) \subset R^n$, where $a(k) \in R_n$, $a(k) \neq 0$, and $w(k)$ is a closed and convex polyhedron determined by

$$A_i x \leqslant b_i,\ i \in c_I(k), \qquad A_i x = b_i,\ i \in c_E(k).$$

For every k,

$$W \subset w(k+1) \subset w(k),$$

$$c_I(k) \subset c_I(k+1) \subset C_I \quad \text{and} \quad c_E(k) \subset c_E(k+1) \subset C_E.$$

The sequence is finite and ends with the original linear pogramming problem.

For the sake of simplicity the argument k will be omitted. The lower level algorithm applied to LLP (with $w \neq \emptyset$) produces a sequence $\{x^{(i)}\} \subset w$ intended to converge to an optimum. The general rules for its construction are

$$x^{(0)} = v^{(1)}, \quad x^{(i)} = v^{(i)} + s^{(i)} d^{(i)} \ \forall i > 0, \tag{1.2}$$

where $v^{(i)} \in w$ is the starting point and $d^{(i)} \in R^n$ is the search direction of the i-th iteration. The real $s^{(i)}$ is determined in the following way:

Let M be the set of all reals $s > 0$ such that

$$\exists j \in \{1, \ldots, m\}: \ A_j(v^{(i)} + sd^{(i)}) = b_j, \ A_j d^{(i)} \neq 0 \tag{1.3}$$

and $\forall \sigma \geqslant 0$: $N(s) \geqslant N(\sigma)$, where $N(\sigma)$ is the number of constraints satisfied at $v^{(i)} + \sigma d^{(i)}$. Then

$$s^{(i)} = \begin{cases} \max M & \text{if } A_0 d^{(i)} < 0 \text{ and } \exists s > 0\colon \ N(0) < N(s), \\ \min M & \text{otherwise.} \end{cases} \tag{1.4}$$

We assume that $c_I \cup c_E$ is precisely the set of all constraints satisfied at $v^{(i)}$. The search direction $d^{(i)}$ is an admissible descent direction, that is,

$$ad^{(i)} < 0 \quad \text{and} \quad \exists s > 0\colon \ v^{(i)} + sd^{(i)} \in w.$$

In the case where $N(s^{(i)}) > N(0)$ we proceed to the next, $(k+1)$-st problem with a new cost vector $a(k+1)$, etc.

Define the relative boundary ∂w of the admissible set w,

$$\partial w = \{x \in w\colon \ \forall \varepsilon > 0 \ \exists z \notin w\colon \ |z - x| < \varepsilon, \ A_i z = b_i \ \forall i \in c_A\},$$

where $c_A = c_E \cup \{i \in c_I\colon \ A_i x = b_i \ \forall x \in w\}$, and the relative interior of w is $w^\circ = w \backslash \partial w$. ∂W and W° are defined in an analogous way. The idea of inner search results from an obvious observation regarding the well-known simplex and reduced gradient methods which also are methods of admissible directions. In simplex methods, every $x^{(i)}$ is a vertex and $[x^{(i)}, x^{(i+1)}]$ is an edge of w. In reduced gradient methods, every $d^{(i)}$ is a projection of a descent direction, e.g., $-a^T$ on w at $v^{(i)}$. In both methods, $v^{(i)} = x^{(i-1)}$ for every i. An important property of both these classes is that the search path, that is, the broken line whose successive vertices are $x^{(i)}$, $i = 0, 1, \ldots$, lies entirely on ∂w (possibly, without its first segment in reduced gradient). Now, suppose that $v^{(i)} \in \partial w$. If we admit $d^{(i)}$ pointing to w°, that is,

$$\exists s > 0\colon \ v^{(i)} + sd^{(i)} \in w^\circ,$$

then generally a greater improvement of the cost can be achieved in the i-th iteration than if the ray $\{x\colon x = v^{(i)} + sd^{(i)}, \ s > 0\}$ lies on ∂w and, in particular, if $d^{(i)}$ is chosen according to simplex or reduced gradient rules. This suggests that methods of admissible directions with search paths which systematically cross w° may have better convergence. To implement this idea it is necessary to impose additional conditions on search directions $d^{(i)}$ and starting points $v^{(i)}$ which guarantee finite convergence and prevent zigzagging. Recall that zigzagging is likely to occur in methods of admissible directions even if finiteness is assured. It then manifests itself in the fact that the steps $|x^{(i+1)}-x^{(i)}|$ are very small. The reason for this phenomenon is in the shape of the admissible set. Let P be the linear subspace of all vectors orthogonal to every A_i, $i \in c_A$. In large real-life problems the dimensions of admissible sets along orthogonal directions from P normally differ by orders of magnitude. In other words, the admissible set resembles a razor blade or a pin rather than a ball. In consequence, the rate of convergence is extremely sensitive to the right choice of search directions. Note also that the most effective search directions are almost parallel to ∂w and very close to the search directions of simplex or reduced gradient.

Among other linear programming algorithms based on admissible search directions which may point to w°, the recent versions of the "adaptive method" of Gabasov and Kirillova ([1], [2]) seem to be most promising. Kallio's method ([4], [5]), which originates from the same idea, is characteristic of earlier attempts in that the measures taken to guarantee finiteness and prevent zigzagging are so severe that practically most of the search directions lie on the relative boundary. This is why its performance is hardly better than that of reduced gradient.

The aim of this paper is to formulate unrestrictive and easily computable conditions on $d^{(i)}$ and $v^{(i)}$ which ensure finite convergence, and on this basis to propose a family of algorithms where inner search directions (i.e., pointing to the relative interior) are systematically used.

2. Auxiliary results. For any x_0, $d \in R^n$ we put

$$P(x_0, d) = \{x \in R^n : x = x_0 + sd,\ s \in R^1,\ s > 0\}.$$

We shall need the following existence result:

LEMMA 1. LP *has an optimal solution if and only if* $W \neq \emptyset$ *and* $P(x, d) \cap W$ *is bounded for any* $x, d \in R^n$ *such that* $A_0 d < 0$.

Proof. Assume that an optimal solution exists. Then $W \neq \emptyset$. Suppose that $P(x,d) \cap W$ is not bounded for some x, d such that $A_0 d < 0$. Then there exists s_m such that

$$x+sd \in W \quad \forall s \geqslant s_m \qquad \text{and} \qquad A_0(x+sd) \to -\infty \text{ as } s \to \infty,$$

which is a contradiction, and so the "only if" part is true. Assume now that $W \neq \emptyset$ and $P(x, d) \cap W$ is bounded for all x, $d \in R^n$ such that $A_0 d < 0$. Let $\pi_1 = R^n$ and $\{x_i^{(1)}\}_{i=1}^{\infty} \subset W \cap \pi_1$ be a sequence such that

$$A_0 x_i^{(1)} \to \inf_{x \in W} A_0 x, \qquad i \to \infty.$$

We distinguish between three cases:

(a) $\{x_i^{(1)}\}$ is bounded. Then it contains a convergent subsequence whose limit is an optimal solution.

(b) $\{x_i^{(1)}\}$ is unbounded and $x_1^{(1)}$ is optimal.
(c) $\{x_i^{(1)}\}$ is unbounded and

$$A_0 x_1^{(1)} > \inf_{x \in W} A_0 x.$$

Without loss of generality we assume that $x_i^{(1)} \neq x_1^{(1)}$ $\forall i > 1$. For each sufficiently large i, $P(x_1^{(1)}, x_i^{(1)} - x_1^{(1)})$ intersects ∂W at some point z_i such that $A_0 z_i \leqslant A_0 x_i^{(1)}$. Hence

$$A_0 z_i \to \inf_{x \in W} A_0 x, \quad i \to \infty.$$

There is at least one $(n-1)$-dimensional hyperplane π_2 in the family of hyperplanes $\pi_1 \cap \{x \in R^n\colon A_i x = b_i\}$, $i \in C_I$, which contains an infinite subsequence $\{z_{k_i}\} \subset \{z_i\}$. Write $\{x_i^{(2)}\} = \{z_{k_i}\}$. Since

$$\{x_i^{(2)}\} \subset W \cap \pi_2 \quad \text{and} \quad A_0 x_i^{(2)} \to \inf_{x \in W} A_0 x, \quad i \to \infty,$$

we can apply the same argument as before to the sequence $\{x_i^{(2)}\}$ to show that either an optimal solution exists or there exists a sequence $\{x_i^{(3)}\} \subset W$ contained in an $(n-2)$-dimensional hyperplane π_3 such that

$$A_0 x_i^{(3)} \to \inf_{x \in W} A_0 x, \quad i \to \infty.$$

This procedure is continued until case (a) or (b) occurs. Since after the k-th step of the procedure we obtain a sequence $\{x_i^{(k+1)}\}$,

$$A_0 x_i^{(k+1)} \to \inf_{x \in W} A_0 x \quad \text{as } i \to \infty,$$

contained in a hyperplane of dimension not greater than $n-k$, we conclude that the maximum number of steps in which case (c) occurs is $n-1$. After less than n steps the procedure therefore yields case (a) (this may happen in the first step) or case (b), which completes the proof.

We say that the j-th constraint is *intersected* by $P(x, d)$ if $A_j x \neq b_j$ and $A_j(x+sd) = b_j$ for some $s > 0$. Then Lemma 1 may be alternatively stated as follows:

LP *has an optimal solution if and only if* $W \neq \emptyset$ *and every ray* $P(x, y-x)$ *such that* $A_0(y-x) < 0$, $x, y \in W$, *intersects some inequality constraint.*

For any matrix H we denote its i-th row by H_i and its j-th column by H^j.

LEMMA 2. *Assume that* B *is a nonsingular* $(n \times n)$*-matrix with the inverse* $D = B^{-1}$. *Let* $\beta \in R^n$, A_j, $A_l \in R_n$ *be given vectors,* $r \in \{1, \ldots, n\}$ *and*

$$x_B = D\beta, \quad y_i = A_i D, \quad y_i^0 = A_i x_B - b_i, \quad i = j, l,$$

for some given reals b_j, b_l. *Define an* $(n \times n)$*-matrix* $\bar{B}$,

$$\bar{B}_i = B_i,\ i = 1, \ldots, n,\ i \neq r, \quad \bar{B}_r = A_j,$$

and a vector $\bar{\beta} \in R^n$,

$$\bar{\beta}_i = \beta_i, \ i = 1, \ldots, n, \ i \neq r, \qquad \bar{\beta}_r = b_j.$$

(i) $\bar{B}$ *is nonsingular if and only if* $y_j^r \neq 0$.

(ii) *Assume that* $\bar{B}$ *is nonsingular with the inverse* $\bar{D}$,

$$\bar{x}_B = \bar{D}\bar{\beta}, \quad \bar{y}_l = A_l\bar{D}, \quad \bar{y}_l^0 = A_l\bar{x}_B - b_l.$$

Then

(2.1) $$\bar{y}_l^r = y_l^r / y_j^r,$$

(2.2) $$\bar{y}_l^i = y_l^i - y_j^i \bar{y}_l^r, \quad i = 0, 1, \ldots, n, \ i \neq r,$$

(2.3) $$\bar{D}^r = D^r / y_j^r,$$

(2.4) $$\bar{D}^i = D^i - y_j^i \bar{D}^r, \quad i = 1, \ldots, n, \ i \neq r,$$

(2.5) $$\bar{x}_B = x_B - y_j^0 \bar{D}^r.$$

Proof. (i) is obvious. (ii) By virtue of the definition,

$$\bar{y}_l\bar{B} = \bar{y}_l B - \bar{y}_l^r B_r + \bar{y}_l^r y_j B.$$

Comparing the coefficients of B_r on both sides of the equality $\bar{y}_l\bar{B} = y_l B$ we get $\bar{y}_l^r y_j^r = y_l^r$. Hence we obtain (2.1). Comparing the coefficients of B_i, $i \neq r$, we get (2.2) for $i > 0$. Formulae (2.3) and (2.4) are similarly derived from the equalities $\bar{D}_i\bar{B} = D_i B$, $i = 1, \ldots, n$. Further,

$$\bar{x}_B = \bar{D}\bar{\beta} = \sum_{i=1}^{n} (D^i - \bar{D}^r y_j^i)\beta_i + \bar{D}^r b_j = x_B - \bar{D}^r (y_j \beta - b_j),$$

whence (2.5). Lastly,

$$\bar{y}_l^0 = A_l \bar{x}_B - b_l = y_l^0 - y_j^0 A_l \bar{D}^r = y_l^0 - y_j^0 \bar{y}_l^r.$$

Let $A_l = B_r$ and $b_l = \beta_r$ in Lemma 2. Since $y_l^r = 1$, $y_l^i = 0$, $i = 0, 1, \ldots, n$, $i \neq r$, we obtain from (2.1) and (2.2) the following

COROLLARY 3. $\bar{y}_l^r = 1/y_j^r$, $\bar{y}_l^i = -y_j^i \bar{y}_l^r$, $i = 0, 1, \ldots, n$, $i \neq r$.

The following generalization of the Farkas theorem belongs to G. F. Voronyi (see [3]). We give it in a different formulation and with a different proof.

LEMMA 4. *Consider a set of vectors* $\{Y_j\}_{j \in S} \subset R_n$, *where* S *is a finite set of indices. Let*

$$S = S_1 \cup S_2 \cup S_3, \quad S_1 \neq \emptyset, \quad S_1 \cap S_2 = S_1 \cap S_3 = S_2 \cap S_3 = \emptyset.$$

The system of inequalities and equations

(2.6) $$Y_j d < 0 \ \forall j \in S_1,$$

(2.7) $$Y_j d \leqslant 0 \ \forall j \in S_2,$$

(2.8) $$Y_j d = 0 \ \forall j \in S_3$$

has a solution $d \in R^n$ *if and only if there exist no reals* $\{\lambda_j\}_{j\in S}$ *such that*

$$\sum_{j\in S} \lambda_j Y_j = 0, \tag{2.9}$$

$$\lambda_j \geqslant 0 \ \forall j \in S_1 \cup S_2, \tag{2.10}$$

$$\sum_{j\in S_1} \lambda_j > 0. \tag{2.11}$$

Proof. Assume that system (2.6)–(2.8) has a solution d. Then

$$\sum_{j\in S} \lambda_j Y_j d < 0$$

for every set of reals $\{\lambda_j\}_{j\in S}$ which satisfy (2.10) and (2.11). Hence (2.9) does not hold. Assume in turn that system (2.6)–(2.8) has no solution. Notice that

$$P_1 = \{d \in R^n\colon\ Y_j d < 0 \ \forall j \in S_1\}$$

is an open convex cone and

$$P_2 = \{d \in R^n\colon\ Y_j d \leqslant 0 \ \forall j \in S_2,\ Y_j d = 0 \ \forall j \in S_3\}$$

is a nonempty closed convex cone. By definition, $P_2 = R^n$ if $S_2 \cup S_3 = \emptyset$. There exists at least one vector $\sigma_2 \in R_n$ supporting with respect to P_2, $\sigma_2 d \leqslant 0 \ \forall d \in P_2$. Every such vector has the form

$$\sigma_2 = \sum_{j\in S_2\cup S_3} \lambda_j Y_j \tag{2.12}$$

for some $\{\lambda_j\}_{j\in S_2\cup S_3}$ such that $\lambda_j \geqslant 0 \ \forall j \in S_2$. Assume $P_1 \neq \emptyset$. Then P_1 has a supporting vector σ_1, $\sigma_1 d < 0 \ \forall d \in P_1$. Every such vector has the form

$$\sigma_1 = \sum_{j\in S_1} \lambda_j Y_j, \quad \lambda_j \geqslant 0 \ \forall j \in S_1,\ \sum_{j\in S_1} \lambda_j > 0. \tag{2.13}$$

It is easy to see that a necessary and sufficient condition for $P_1 \cap P_2 = \emptyset$ is that there exist supporting vectors σ_1 of the form (2.13) and σ_2 of the form (2.12) such that $\sigma_1 + \sigma_2 = 0$, which gives (2.9)–(2.11). If $P_1 = \emptyset$, then there exists a linear combination (2.13) of the vectors Y_j, $j \in S_1$, such that $\sigma_1 = 0$. We take $\sigma_2 = 0$ and again have $\sigma_1 + \sigma_2 = 0$, that is, (2.9)–(2.11).

A concept of dual admissibility plays an important role in inner search methods. For the problem LLP, we call a set of constraints $E \subset c_I \cup c_E$ *dually admissible* if there exist reals λ_i, $i \in E$, such that

$$a = \sum_{i\in E} \lambda_i A_i, \quad \lambda_i \leqslant 0 \ \forall i \in E \cap c_I. \tag{2.14}$$

Putting $a = A_0$, $c_I = C_I$ and $c_E = C_E$ we obtain an analogous definition for LP. To characterize this concept in terms more directly connected with search directions we define the cone of descent directions

(2.15) $$K_0 = \{d \in R^n: \; ad < 0\}$$

and the cone of directions which are admissible with respect to every constraint from E,

$$K_E = \{d \in R^n: \; A_i d \leqslant 0 \;\; \forall i \in E \cap c_I, \; A_i d = 0 \;\; \forall i \in E \cap c_E\}.$$

LEMMA 5. *$K_0 \cap K_E = \emptyset$ if and only if E is dually admissible.*

Proof. $K_0 \cap K_E = \emptyset$ if and only if the system

(2.16) $$ad < 0, \quad A_i d \leqslant 0 \;\; \forall i \in E \cap c_I, \quad A_i d = 0 \;\; \forall i \in E \cap c_E$$

has no solution d. Let

$$S_1 = \{0\}, \quad S_2 = E \cap c_I, \quad S_3 = E \cap c_E,$$

$$Y_0 = a, \quad Y_i = A_i \;\; \forall i \in S_2 \cup S_3, \quad S = S_1 \cup S_2 \cup S_3.$$

By virtue of Lemma 4, (2.16) has no solution if and only if there are reals λ_i, $i \in S$, such that

$$\lambda_0 a + \sum_{i \in S_2 \cup S_3} \lambda_i A_i = 0, \quad \lambda_0 > 0, \; \lambda_i \geqslant 0 \;\; \forall i \in S_2,$$

which is equivalent to the dual admissibility of E.

A set of constraints $E \subset C_I \cup C_E$ is called *independent* if the vectors A_i, $i \in E$, are linearly independent.

The next lemma describes relationships between dually admissible sets of constraints and optimal solutions.

LEMMA 6. (i) *Assume that E is a dually admissible set of constraints in the problem* LP *and $x_B \in R^n$ satisfies $A_i x_B = b_i \;\; \forall i \in E$. Then*

$$A_0 x_B \leqslant A_0 x \;\; \forall x \in W.$$

(ii) *Assume that $\hat{x}$ is an optimal solution of* LP. *Then*

$$E = \{i \in C_I \cup C_E: \; A_i \hat{x} = b_i\}$$

is dually admissible.

(iii) *Assume that $W \neq \emptyset$. Then* LP *has an optimal solution if and only if $C_I \cup C_E$ is dually admissible.*

Proof. (i) For every $x \in W$ we have

$$A_i(x_B - x) \geqslant 0 \;\; \forall i \in E \cap C_I \quad \text{and} \quad A_i(x_B - x) = 0 \;\; \forall i \in E \cap C_E.$$

Moreover,

$$A_0 = \sum_{i \in E} \lambda_i A_i, \quad \lambda_i \leqslant 0 \;\; \forall i \in E \cap C_I.$$

Thus

$$\sum_{i\in E} \lambda_i A_i(x_B - x) \leqslant 0 \quad \text{and} \quad A_0(x_B - x) \leqslant 0.$$

(ii) is evident, since if E is not dually admissible, then there exists $d \in K_0 \cap K_E$ (Lemma 5) such that $x + sd \in W$ for some $s > 0$, which is a contradiction.

(iii) Let $C_I \cup C_E$ be dually admissible. It thus contains a dually admissible and independent subset E. Let x_B satisfy $A_i x_B = b_i\ \forall i \in E$. By virtue of (i),

$$A_0 x_B \leqslant A_0 x \ \forall x \in W.$$

Then every ray $P(x, d)$, $d \in K_0$, has a bounded intersection with W, and so an optimal solution exists (see Lemma 1). The "only if" part follows from (ii).

To characterize more fully the situation in Lemma 6 (i) we state the following obvious lemma:

LEMMA 7. *Let* $E \subset \{1, \ldots, m\}$. *Assume that*

$$A_i x_B = b_i,\ A_i x'_B = b_i \ \forall i \in E \quad \textit{and} \quad A_0 = \sum_{i\in E} \lambda_i A_i.$$

Then $A_0 x_B = A_0 x'_B$.

3. General characterization of inner search algorithms. In this section we formulate the principles of inner search which together with the features described in Section 1 will enable us to discuss convergence in the next section. Since the upper level algorithm is not specific for inner search and can be constructed as in other methods of admissible directions, we only recall that the initial starting point $v^{(1)}$ of LLP$(k+1)$ is the last point $x^{(q)}$ of LLP(k) and the cost vector in LLP(k) is given by

$$a = \sum_{i\in P} \alpha_i A_i,$$

where

$$P = \{0, 1, \ldots, m\} \backslash (c_I \cup c_E),$$

$$\alpha_0 \geqslant 0 \quad \text{and} \quad \alpha_i (A_i v^{(1)} - b_i) \geqslant 0 \ \forall i \in P \backslash \{0\}.$$

Moreover, $aA_i^T \neq 0$ for at least one $i \in P\backslash\{0\}$ if $P \neq \{0\}$.

The lower level algorithm is the basic one; it includes the distinctive features of inner search. In every iteration of this algorithm a set of constraints $E^{(i)} \subset c_I \cup c_E$ is determined, which contains (among others) all constraints active at $v^{(i)}$. Together with $E^{(i)}$ we determine the basic set $E_B^{(i)} \subset E^{(i)}$. It is independent and contains (among others) the maximum number of constraints active at $v^{(i)}$ for which this property holds. Any vector $x_B^{(i)}$ which satisfies $A_j x_B^{(i)} = b_j\ \forall j \in E_B^{(i)}$ is called a *basic solution*. $K_E^{(i)}$ denotes the cone of all directions admissible with respect to the constraints in $E^{(i)}$,

(3.1) $$K_E^{(i)} = \{d \in R^n\colon A_j d \leqslant 0\ \forall j \in E^{(i)} \cap c_I,\ A_j d = 0\ \forall j \in E^{(i)} \cap c_E\},$$

and K_0 is the cone of descent directions (2.15).

The lower level algorithm is divided into two parts, A and B. Part A may be briefly characterized as the search for a dually admissible basic set of constraints, and Part B as the search for optimum by means of admissible directions combined with dual simplex. The algorithm of Part A has the following properties, fundamental for finite convergence. By J_A we denote the set of all iteration numbers of Part A.

H1. $\exists c < 0\ \forall\{i, i+1\} \subset J_A$: $a(x^{(i+1)} - x^{(i)}) \leqslant c|x^{(i+1)} - x^{(i)}|$.

If the simplest rule for the construction of starting points is at work,

$$v^{(i+1)} = x^{(i)}\ \forall\{i, i+1\} \subset J_A,$$

then H1 results from the following:

$$\exists c < 0\ \forall i \in J_A:\ ad^{(i)} \leqslant c|d^{(i)}|.$$

H2. In every k successive iterations, say $i, i+1, \ldots, i+k-1$, where k is a predetermined positive integer, a constraint which does not belong to $E^{(i)}$ must be intersected, that is,

$$\forall\{i, i+1, \ldots, i+k-1\} \subset J_A\ \exists j \in \{i, i+1, \ldots, i+k-1\}\ \exists l \in c_I \backslash E^{(i)}:\ A_l x^{(j)} = b_l.$$

To obtain this intersection it is sufficient that $d^{(i)} \in K_0 \cap K_E^{(i)}$.

H3. If $A_j x^{(i)} = b_j$ for some $j \in c_I \cup c_E$ and $\{i, i+1\} \subset J_A$, then $j \in E^{(i+1)}$.

Denote by $\alpha^{(i)}$ the number of constraints from the set $c_I \backslash E^{(i)}$ which are active at $x^{(i)}$.

H4. If $j \notin E^{(i-1)}$ and $j \in E^{(i)}$ for some $\{i-1, i\} \subset J_A$, then $j \in E^{(l)}\ \forall l \in J_A$ such that $l > i$ and

$$\sum_{t=i}^{l-1} \alpha^{(t)} < r;$$

r is a positive constant independent of i.

Part A ends in the following situations.

(i) $K_0 \cap K_E^{(i)} = \emptyset$; this is the basic stopping condition. It means that a dually admissible set of constraints $E^{(i)}$ has been found. The algorithm passes over to Part B.

(ii) The ray $P(v^{(i)}, d^{(i)})$ does not intersect any constraint, that is, rules (1.2)–(1.4) yield no point $x^{(i)}$. This may happen only if $w = W \neq \emptyset$ and LP has no optimal solution.

(iii) Rules (1.2)–(1.4) give a point $x^{(i)}$ admissible with respect to some constraint which is not satisfied at $v^{(i)}$, that is, $N(s^{(i)}) > N(0)$. The upper level algorithm is then activated.

Note that $E^{(i)}$ is dually admissible if and only if it contains a dually admissible basic set $E_B^{(i)}$. Typically, in computational realizations of the algorithms of Part A we have $E^{(i)} = E_B^{(i)}$. As will be shown, in this case it is relatively easy to verify the stopping condition (i).

In every iteration of Part B the condition of dual admissibility $K_0 \cap K_E^{(i)} = \emptyset$ is satisfied. The sets $E_B^{(i)}$ are dually admissible. The search directions are admissible descent directions defined by

$$d^{(i)} = x_B^{(i)} - v^{(i)}, \tag{3.2}$$

where the starting point $v^{(i)} \in w$ is not an optimal solution. Apart from the sequence of admissible solutions $x^{(i)} \in w$, Part B produces a sequence of basic solutions $x_B^{(i)}$ such that the corresponding cost values $ax_B^{(i)}$ monotonically increase. The algorithm ends in one of the following situations:

(i) The basic solution is admissible, $x_B^{(i)} \in w$. Then $x_B^{(i)}$ is an optimal solution of the lower level problem LLP. If $w = W$, then it is also an optimal solution of LP. In the case where $w \neq W$, further procedure depends on whether every constraint satisfied at $x_B^{(i)}$ is satisfied at $v^{(i)}$. If so, and $\alpha_0 = 0$ in the formula for a, then $W = \emptyset$. Otherwise, the algorithm passes over to the upper level.

(ii) As (iii) for Part A.

If neither of these situations occurs, the constraint (or constraints) intersected at $x^{(i)}$ is introduced into the basic set $E_B^{(i)}$, usually in place of one (or more, respectively) of its elements. The exchange is performed in such a way that the dual admissibility of the next basic set $E_B^{(i+1)}$ is maintained (for details and proofs see Section 5).

4. Convergence of Part A. It will be shown that properties H1–H4 formulated in Section 3 implicate finite convergence of Part A provided an optimal solution exists. It will also be shown how to guarantee finiteness without this assumption.

LEMMA 8. *Assume that* LLP *has an optimal solution and a sequence* $\{x^{(i)}\} \subset w$ *satisfies* H1. *Then* $\{x^{(i)}\}$ *is either finite or convergent to some element of* w.

Proof. Assume that $\{x^{(i)}\}$ is infinite. By virtue of H1,

$$ax^{(i+1)} \leqslant ax^{(i)} \ \forall i.$$

Since an optimal solution exists, the sequence $\{ax^{(i)}\}$ is lower bounded, and so it is convergent. From H1 we obtain

$$|x^{(j)} - x^{(i)}| \leqslant \frac{1}{c}(ax^{(j)} - ax^{(i)}) \ \forall i,j, \ i \leqslant j.$$

As $\{ax^{(i)}\}$ is convergent, $\{x^{(i)}\}$ is a Cauchy sequence in a closed set w. Thus it is convergent in w, which completes the proof.

Below we give an estimate on the number of constraints active at the limit point of the sequence $\{x^{(i)}\}$ produced by a Part A algorithm. The theorem is formulated in terms of the lower level problem LLP.

THEOREM 9. *Let $\{x^{(i)}\} \subset w$ be an infinite sequence convergent to some z and let $\{E^{(i)}\}$ be an infinite sequence of subsets of $c_I \cup c_E$. Assume* H2, H3, H4 *with J_A equal to the set of all positive integers. Then the number of constraints from $c_I \backslash c_A$ active at z is greater than r.*

Proof. Suppose that the j-th constraint is active at $x^{(i)}$ and $j \in c_I \backslash E^{(i)}$. Such i and j exist by virtue of H2. Thus $j \in E^{(i+1)}$ (by H3) and may not leave the sets $E^{(l)}$, $l > i+1$, until r constraints different from j have been active (by H4). This happens not later than after kr iterations (by H2). Since this reasoning may be repeated for every constraint, these r constraints are also different from each other. Thus, the number of constraints from $c_I \backslash c_A$ with the property

$$\forall i \, \exists j > i \colon \; A_l x^{(j)} = b_l$$

is greater than r. To complete the proof let us notice that if a constraint is active at an infinite number of points $x^{(i)}$, then it is active at z.

The next theorem gives sufficient conditions of finite convergence of algorithms of Part A under the assumption that an optimum exists. Notice that the requirement $d^{(i)} \in K_0$ is not directly engaged in the proof. By p we denote the maximum number of constraints from $c_I \backslash c_A$ simultaneously active at some point of w.

THEOREM 10. *Let $\{x^{(i)}, E^{(i)}\}$ be a sequence of pairs such that $x^{(i)} \in w$ and $E^{(i)} \subset c_I \cup c_E$ for every i. Assume that* LLP *has an optimal solution and* H1–H4 *hold with $r \geqslant p$. Then the sequence $\{x^{(i)}\}$ is finite.*

Proof. Suppose that $\{x^{(i)}\}$ is infinite. It follows from Lemma 8 that $\{x^{(i)}\}$ is convergent to some $z \in w$. By Theorem 9 the number of constraints from $c_I \backslash c_A$ active at z is greater than r, which is a contradiction.

We call a linear problem *nondegenerate* if the number of simultaneously active constraints, both of equality and inequality type, does not exceed n. In such problems it is sufficient to keep n constraints in the sets $E^{(i)}$ to assure finiteness of Part A, provided an optimal solution exists.

Let $\{x^{(i)}, E^{(i)}\}_{i=1}^{q}$ be a sequence such that $x^{(i)} \in w$ and $E^{(i)} \subset c_I \cup c_E$ for every i, satisfying H1–H4. We say that this sequence is *continuable* if there exist $x^{(q+i)}$, $E^{(q+i)}$, $i = 1, \ldots, k$, such that the sequence $\{x^{(i)}, E^{(i)}\}_{i=1}^{q+k}$ fulfils the same assumptions.

THEOREM 11. *Let* LLP *have an optimal solution and the sequence $\{x^{(i)}, E^{(i)}\}_{i=1}^{q}$ satisfy all the assumptions in the definition of continuability. If this sequence is not continuable, then every set $E^{(q+1)} \subset c_I \cup c_E$ such that $\{E^{(i)}\}_{i=1}^{q+1}$ satisfies* H3 *and* H4 *is dually admissible.*

Proof. Every constraint active at $x^{(q)}$ belongs to $E^{(q+1)}$. If $x^{(q)}$ is optimal, then $E^{(q+1)}$ is dually admissible by Lemma 6 (ii). Assume that $x^{(q)}$ is not optimal and $E^{(q+1)}$ is not dually admissible. Then there exists a search direction $d^{(q+1)} \in K_0 \cap K_E^{(q+1)}$ such that the ray $P(x^{(q)}, d^{(q+1)})$ intersects a constraint from $c_I \backslash E^{(q+1)}$ at some point $x^{(q+1)} \in \partial w$ (by Lemma 1). Hence the sequence $\{x^{(i)}, E^{(i)}\}_{i=1}^{q}$ is continuable. This contradiction completes the proof.

The converse is not true: a sequence $\{x^{(i)}, E^{(i)}\}_{i=1}^{q}$ such that $x^{(i)} \in w$, $E^{(i)} \subset c_I \cup c_E$ for every i may be in accordance with H1–H4, even if $E^{(q)}$ is dually admissible. However, in the case where $K_0 \cap K_E^{(i)} = \emptyset$ Part A is stopped by condition (i) of Section 3. This is justified by the fact that in this situation it may be difficult to find a search direction, if one exists, which would make the continuation of Part A possible. Recall that as long as $K_0 \cap K_E^{(i)} \neq \emptyset$ we may always choose $d^{(i)} \in K_0 \cap K_E^{(i)}$ and obtain $x^{(i)} \in \partial w$ such that $A_l x^{(i)} = b_l$ for some $l \in c_I \backslash E^{(i)}$. Besides, a more effective procedure of Part B may then be begun. To end this discussion let us note that it is possible to construct a sequence $\{x^{(i)}, E^{(i)}\}_{i=1}^{q}$ satisfying all the above assumptions, where $E^{(q)}$ is dually admissible and exclusively consists of constraints which were active in previous iterations.

We shall now consider the case where $w = W \neq \emptyset$, $a = A_0$ and no optimal solution exists. This is equivalent to

(4.1) $$\exists x, d \in R^n\colon\ ad < 0 \text{ and } \forall s \geqslant 0,\ x + sd \in w.$$

Under this assumption it is evident that even if a sequence of pairs $\{x^{(i)}, E^{(i)}\}$ satisfies all the assumptions of Theorem 10, it may be infinite. Moreover, if it is infinite, then it does not converge and there exists more than one constraint $l \in c_I$ such that

(4.2) $$\forall i\ \exists j > i\ \exists k > i\colon\ A_l x^{(j)} = b_l \text{ and } A_l x^{(k)} \neq b_l.$$

The finiteness of Part A in the general case where one does not know whether an optimal solution exists is guaranteed by a special reduced-gradient subalgorithm, called the *checking algorithm*. The checking algorithm can be started after any iteration of Part A; its starting point $v^{(1)}$ is the last point $x^{(q)}$ obtained in Part A. For every iteration number $i > 1$ we have $v^{(i)} = x^{(i-1)}$, the search direction $d^{(i)}$ is an admissible descent direction and

(4.3) $$A_l d^{(i)} = 0\ \ \forall l \in c_I \cup c_E \text{ such that } A_l v^{(i)} = b_l.$$

$x^{(i)}$ is computed according to (1.2)–(1.4). Notice that in the case where $w = W$ we have

$$s^{(i)} = \max\{s \geqslant 0\colon\ v^{(i)} + sd^{(i)} \in W\}.$$

The stopping conditions of Part A remain valid, although condition (i) of Section 3 may be omitted and condition (iii) is necessary only if the checking algorithm is used in the case $w \neq W$. The checking algorithm is also terminated if system (4.3) has no solution $d^{(i)}$ which is an admissible descent direction. In this case we return to the main algorithm of Part A. It can be restarted from the last point $x^{(q)}$ obtained in the checking algorithm, $v^{(1)} = x^{(q)}$, with the set $E^{(1)}$ consisting of all constraints active at $v^{(1)}$.

The checking algorithm is finite and the number of its iterations is less than n. This follows from the fact that the number of linearly independent equations (4.3) strictly increases in every iteration.

We say that the checking algorithm is *successful* if in one of its iterations $P(v^{(i)}, d^{(i)}) \subset w$. To formulate sufficient conditions for that we define

$$Z_l(s) = \{x \in w\colon ax \leqslant s \text{ and } A_l x = b_l\} \ \forall l \in c_I \cup c_E.$$

It is evident that under assumption (4.1) there exist reals s_m such that, for every $s < s_m$ and every l, $Z_l(s)$ is unbounded if it is nonempty. Let s^* be the supremum of all such reals s_m. A sufficient condition for the checking algorithm to be successful, provided (4.1) is fulfilled, is

$$v^{(1)} \in w \quad \text{and} \quad av^{(1)} < s^*. \tag{4.4}$$

For the finiteness of Part A in the general case it suffices that the checking algorithm is repeated after every K iterations of Part A, where K is an arbitrary predetermined constant. The criterion for starting the checking algorithm may also be based on property (4.2). To see that this also guarantees finiteness, consider an algorithm of Part A producing a sequence $\{x^{(i)}, E^{(i)}\}$ for which all the assumptions of Theorem 10 are satisfied (of course, except for the existence of optimal solution). By virtue of (4.2) we may suspect that there is no optimal solution if we observe that a constraint re-enters the set $E^{(i)}$. The more re-entries of the same constraints, the more chances that no optimal solution exists and the sufficient condition (4.4) is fulfilled.

5. Continuability and finiteness of Part B. Let us begin with a few simple consequences of the basic definitions given in Section 3. First, Part B may be applied to an LLP only if this problem has an optimal solution. This follows from the assumption $K_0 \cap K_E^{(i)} = \emptyset$ for every i and Lemma 6 (iii). By Lemma 6 (i) the search directions (3.2) satisfy $ad^{(i)} < 0$. By Lemma 1, for every i there exists a unique point $x^{(i)}$ determined by rules (1.2)–(1.4). We shall consider the constraints active at $x^{(i)}$ in more detail. An arbitrary constraint $j \in E_B^{(i)} \cap c_I$, nonactive at $v^{(i)}$, is active at $x^{(i)}$ if and only if $x^{(i)}$ is an optimal basic solution, that is,

$$A_l x^{(i)} = b_l \ \forall l \in E_B^{(i)}, \qquad x^{(i)} \in w \text{ and } ax^{(i)} \leqslant ax \ \forall x \in w.$$

This follows evidently from the fact that if a constraint from $E_B^{(i)}$ is nonactive at $v^{(i)}$, then $P(v^{(i)}, d^{(i)})$ intersects it at the basic solution $x_B^{(i)}$. Furthermore, we have the following lemma, essential for the continuability of algorithms of Part B.

LEMMA 12. *Assume that, in the i-th iteration of Part B,* $A_j x^{(i)} = b_j$ *and* $A_j v^{(i)} < b_j$ *for some* $j \in c_I$. *Let*

$$A_j = \sum_{l \in E_B^{(i)}} \lambda_{jl}^{(i)} A_l.$$

Then $\lambda_{jl}^{(i)} > 0$ *for some* $l \in E_B^{(i)}$, *nonactive at* $v^{(i)}$.

Proof. Evidently, $A_j d^{(i)} > 0$. Hence

$$\sum_{l \in E_B^{(i)}} \lambda_{jl}^{(i)} A_l d^{(i)} > 0.$$

Since $A_l x_B^{(i)} = b_l$ and $A_l v^{(i)} \leqslant b_l$, we obtain $A_l d^{(i)} \geqslant 0 \ \forall l \in E_B^{(i)}$. This completes the proof.

In every iteration of Part B a dually admissible and independent set of constraints $E_B^{(i)}$ is needed. We shall now discuss the conditions of dual admissibility of $E_B^{(i+1)}$ provided $E_B^{(i)}$ is dually admissible. Assume first that A_j is linearly independent of the vectors A_l, $l \in E_B^{(i)}$, $E_B^{(i)}$ is dually admissible and $E_B^{(i+1)} = E_B^{(i)} \cup \{j\}$. Then, of course, $E_B^{(i+1)}$ is also dually admissible. In the more typical case where the vector A_j of the constraint which enters $E_B^{(i+1)}$ is linearly dependent on the vectors A_l, $l \in E_B^{(i)}$, there must be a constraint k simultaneously leaving this set:

$$E_B^{(i+1)} = E_B^{(i)} \backslash \{k\} \cup \{j\}, \qquad k \in E_B^{(i)}.$$

For our purposes it is sufficient to consider these two cases.

LEMMA 13. *Assume that* $E_B^{(i)}$ *is dually admissible and independent and*

$$E_B^{(i+1)} = E_B^{(i)} \backslash \{k\} \cup \{j\} \quad \textit{for some } k \in E_B^{(i)}.$$

Let also

$$A_j = \sum_{l \in E_B^{(i)}} \lambda_{jl}^{(i)} A_l, \qquad \lambda_{jk}^{(i)} \neq 0. \tag{5.1}$$

Put

$$a = \sum_{l \in E_B^{(i)}} \lambda_l^{(i)} A_l = \sum_{l \in E_B^{(i+1)}} \lambda_l^{(i+1)} A_l. \tag{5.2}$$

(i) *If* $\lambda_k^{(i)} = 0$, *then*

$$\lambda_j^{(i+1)} = 0, \qquad \lambda_l^{(i+1)} = \lambda_l^{(i)} \ \forall l \in E_B^{(i+1)} \backslash \{j\}$$

and $E_B^{(i+1)}$ *is dually admissible.*

(ii) *Assume* $\lambda_k^{(i)} < 0, j, k \in c_I$. *Then* $E_B^{(i+1)}$ *is dually admissible if and only if* $\lambda_{jk}^{(i)} > 0$ *and*

$$\lambda_k^{(i)}/\lambda_{jk}^{(i)} \geqslant \lambda_l^{(i)}/\lambda_{jl}^{(i)} \ \forall l \in E_B^{(i)} \cap c_I \ \textit{such that } \lambda_{jl}^{(i)} > 0. \tag{5.3}$$

(iii) *Assume* $\lambda_k^{(i)} < 0$, $j \in c_E$, $k \in c_I$. *Then* $E_B^{(i+1)}$ *is dually admissible if and only if*

$$\lambda_l^{(i)}/\lambda_k^{(i)} \geqslant \lambda_{jl}^{(i)}/\lambda_{jk}^{(i)} \ \forall l \in E_B^{(i)} \cap c_I.$$

Proof. From Lemma 2 we obtain, after an appropriate change of notation,

$$\lambda_j^{(i+1)} = \lambda_k^{(i)}/\lambda_{jk}^{(i)}, \qquad \lambda_l^{(i+1)} = \lambda_l^{(i)} - \lambda_{jl}^{(i)} \lambda_j^{(i+1)} \ \forall l \in E_B^{(i+1)} \backslash \{j\}.$$

(i) is obvious.

(ii) $\lambda_{jk}^{(i)} > 0$ is necessary and sufficient for $\lambda_j^{(i+1)} \leqslant 0$. Under the assumption $\lambda_{jk}^{(i)} > 0$, (5.3) is a necessary and sufficient condition for

$$\lambda_l^{(i+1)} \leqslant 0 \ \forall l \in E_B^{(i+1)} \cap c_I.$$

(iii) is proved by a similar argument with the difference that $\lambda_j^{(i+1)} \leqslant 0$ is not required.

Normally, only cases (i) and (ii) of the above lemma may occur in algorithms of Part B; case (iii) is useful for some modifications where the set $c_I \cup c_E$ is not constant throughout the algorithm. Let us note that, due to Lemmas 12 and 13, Part B can be continued until the stopping conditions of Section 3 are fulfilled.

LEMMA 14. *Assume that $E_B^{(i)}$ is dually admissible and independent,*

$$E_B^{(i+1)} = E_B^{(i)} \setminus \{k\} \cup \{j\} \quad \textit{for some } k \in E_B^{(i)}$$

and assumptions (5.1) *and* (5.2) *hold. Let*

$$A_l x_B^{(i)} = b_l \ \forall l \in E_B^{(i)} \quad \textit{and} \quad A_l x_B^{(i+1)} = b_l \ \forall l \in E_B^{(i+1)}.$$

(i) *If $\lambda_k^{(i)} = 0$, then $ax_B^{(i+1)} = ax_B^{(i)}$.*

(ii) *Assume that $\lambda_k^{(i)} < 0$, $j, k \in c_I$ and $E_B^{(i+1)}$ is dually admissible. Let also $A_j x_B^{(i)} \geqslant b_j$. Then*

$$ax_B^{(i+1)} \geqslant ax_B^{(i)}.$$

Moreover, $ax_B^{(i+1)} = ax_B^{(i)}$ if and only if $A_j x_B^{(i)} = b_j$.

Proof. From (2.2) of Lemma 2 we obtain, after the substitution $A_l = a$,

$$ax_B^{(i+1)} = ax_B^{(i)} + \lambda_j^{(i+1)}(b_j - A_j x_B^{(i)}). \tag{5.4}$$

This and Lemma 13 easily yield the assertion.

The assumption $A_j x_B^{(i)} \geqslant b_j$ is obviously satisfied in every iteration of Part B, since $j \in c_I$ and $v^{(i)}$ is admissible with respect to every constraint from c_I.

In every iteration of Part B we obtain an upper and a lower estimate for the optimal cost $a\hat{x}$ in a natural way:

$$ax_B^{(i)} \leqslant a\hat{x} \leqslant \min_{l \leqslant i} ax^{(l)},$$

where $x_B^{(i)}$ is the current basic solution, $x^{(l)}$ is the intersection point of $P(v^{(l)}, d^{(l)})$ with ∂w. If there are many basic solutions $x_B^{(i)}$, all of them yield the same value of cost, as can be seen from Lemma 7. Due to Lemma 14, both estimates monotonically improve in the course of Part B, though $ax^{(l)}$ monotonically decreases only under special assumptions on the choice of $v^{(l)}$, for example, if $v^{(l)} = x^{(l-1)}$ for every l.

In order to formulate the theorem on finite convergence of the algorithm of Part B we need the definition of dual nondegeneracy. We call LLP *dually nondegenerate* if for every dually admissible and independent set $E_B \subset c_I \cup c_E$ we have

$$\lambda_l < 0 \ \forall l \in E_B \cap c_I.$$

The reals λ_l are defined as in (2.14).

THEOREM 15. *Assume that the problem* LLP *is dually nondegenerate and* $\{x_B^{(i)}\}$ *is the sequence of all basic solutions produced by the algorithm of Part B. Then this sequence is finite. Moreover, if* $w = W$, *then its last element is an optimal solution of* LP.

Proof. The number of dually admissible sets $E_B \subset c_I \cup c_E$ is finite. Due to Lemma 13 and the assumption of dual nondegeneracy we have $\lambda_j^{(i+1)} < 0$ in formula (5.4) for every i. Hence the lower estimate of the optimal cost $ax_B^{(i)}$ strictly increases, which completes the proof.

Let us note that this theorem is valid irrespective of the particular choice of $v^{(i)}$. It is difficult to give a good upper estimate of the number of iterations of Part B which are needed to obtain an optimal solution. Of course, it is not greater than the number of all dually admissible and independent sets of constraints. Notice that the assumption of dual nondegeneracy in Theorem 15 is not crucial for the finiteness of Part B. However, if this assumption is removed, more complicated rules of exchange of the sets $E_B^{(i)}$ are necessary to guarantee it. The tools which can be used in order to overcome degeneracy are analogous to those in simplex methods (like lexicographic ordering).

6. Connections between Part B and dual simplex. The algorithm of Part B has strong connections with dual simplex algorithms. Due to this fact it is possible in each iteration to switch from Part B to dual simplex. Let us recall the principles of the latter using a formulation appropriate for LLP. Assume that in the i-th iteration we have a dually admissible, independent set of constraints $E_B^{(i)} \subset c_I \cup c_E$ and a basic solution $x_B^{(i)}$ defined by

$$A_l x_B^{(i)} = b_l \quad \forall l \in E_B^{(i)}.$$

If $x_B^{(i)} \in w$, then it is optimal (see Lemma 6 (i)). Assume that this is not the case. A new dually admissible, independent set $E_B^{(i+1)}$ is then created together with the corresponding basic solution $x_B^{(i+1)}$ in such a way that $ax_B^{(i+1)} \geqslant ax_B^{(i)}$. A constraint $j \in (c_I \cup c_E) \setminus E_B^{(i)}$ enters $E_B^{(i+1)}$. If A_j is linearly independent of A_l, $l \in E_B^{(i)}$, then

$$E_B^{(i+1)} = E_B^{(i)} \cup \{j\}.$$

Otherwise,

$$E_B^{(i+1)} = E_B^{(i)} \setminus \{k\} \cup \{j\} \qquad \text{for some } k \in E_B^{(i)}.$$

The condition of linear independence of the vectors A_l, $l \in E_B^{(i+1)}$, is given in Lemma 2 (i) and the conditions for $E_B^{(i+1)}$ to be dually admissible are in Lemma 13. Lemma 14 gives the relationships between the cost values $ax_B^{(i+1)}$ and $ax_B^{(i)}$. In consequence, a dual simplex iteration can be performed if and only if one of the following situations occurs:

(i) There exists a constraint $j \in c_I \cup c_E$ such that A_j is linearly independent of A_l, $l \in E_B^{(i)}$. Then $E_B^{(i+1)} = E_B^{(i)} \cup \{j\}$ is dually admissible and $ax_B^{(i+1)} = ax_B^{(i)}$.

(ii) There exist $j \in (c_I \cup c_E)\backslash E_B^{(i)}$ and $k \in c_I \cap E_B^{(i)}$ such that $\lambda_{jk}^{(i)} \neq 0$ and $\lambda_k^{(i)} = 0$. Then $E_B^{(i+1)} = E_B^{(i)}\backslash\{k\} \cup \{j\}$ is dually admissible and independent, and $ax_B^{(i+1)} = ax_B^{(i)}$.

(iii) There exist $j \in c_I\backslash E_B^{(i)}$ and $p \in c_I \cap E_B^{(i)}$ such that $A_j x_B^{(i)} \geqslant b_j$, $\lambda_{jp}^{(i)} > 0$ and $\lambda_l^{(i)} < 0$ $\forall l \in c_I \cap E_B^{(i)}$ such that $\lambda_{jl}^{(i)} > 0$. Then

$$E_B^{(i+1)} = E_B^{(i)}\backslash\{k\} \cup \{j\},$$

where k is chosen according to Lemma 13 (ii), is dually admissible and independent. Moreover, $ax_B^{(i+1)} > ax_B^{(i)}$ if $A_j x_B^{(i)} > b_j$ and $ax_B^{(i+1)} = ax_B^{(i)}$ if $A_j x_B^{(i)} = b_j$.

(iv) There exist $j \in c_E\backslash E_B^{(i)}$ and $p \in c_I \cap E_B^{(i)}$ such that $\lambda_{jp}^{(i)} \neq 0$ and $\lambda_l^{(i)} < 0$ $\forall l \in c_I \cap E_B^{(i)}$ such that $\lambda_{jl}^{(i)}$ is of the same sign as $\lambda_{jp}^{(i)}$. Then

$$E_B^{(i+1)} = E_B^{(i)}\backslash\{k\} \cup \{j\},$$

where k is chosen according to Lemma 13 (iii), is dually admissible and independent.

If none of these situations takes place, then $w = \emptyset$. It is important that for every constraint which is not satisfied at $x_B^{(i)}$ we have the following alternative: either this constraint can enter the set $E_B^{(i+1)}$ in the $(i+1)$-st iteration or $w = \emptyset$. Of course, in one iteration there may be many possible exchanges of the set $E_B^{(i)}$ as the situations (i)–(iv) do not exclude each other. To improve convergence, the following list of priorities is recommended:

1. (i) and (iv);
2. (iii) with $A_j x_B^{(i)} > b_j$;
3. (ii) and (iii) with $A_j x_B^{(i)} = b_j$.

If LLP is dually nondegenerate, then any dual simplex algorithm based on the above principles is finite. In the case where $\lambda_k^{(i)} = 0$ (dual degeneracy) there is a possibility of cycling which may lead to the lack of convergence. There are however well-known tools, like lexicographic ordering, to overcome this difficulty and assure finiteness.

It is now evident that an iteration of Part B may be regarded as an iteration of dual simplex with a particular method of determining j, the constraint entering the basic set $E_B^{(i)}$. Notice that almost every dual simplex rule for the determination of j is equivalent to some choice of starting points $v^{(i)}$ in Part B, which means that the effect is identical. On the other hand, Part B is an algorithm of admissible directions and in each iteration gives more information than dual simplex. In particular, it yields an admissible solution $x^{(i)} \in \partial w$ and a two-sided, monotonically improving estimate of the optimal cost. Of course, an iteration of Part B requires more computations than an iteration of dual simplex.

7. Some computational questions. The previous sections are devoted to the presentation of inner search principles. Below we give a brief discussion of several more detailed questions, vital for computational efficiency.

7.1. *Variable system of coordinates and recursive formulae.* Let $E_B^{(i)}$ be the basic set of constraints in the i-th iteration of an inner search algorithm. By $B^{(i)}$ we denote the basic matrix. This is any nonsingular matrix which has all vectors A_l, $l \in E_B^{(i)}$, as its rows. By $u(l)$, $l \in E_B^{(i)}$, we denote the position of the vector A_l in $B^{(i)}$, that is, $B_{u(l)}^{(i)} = A_l$. We also determine the vector $\beta^{(i)} \in R^n$ such that

$$\beta_{u(l)}^{(i)} = b_l \ \forall l \in E_B^{(i)}.$$

The remaining elements of $\beta^{(i)}$ are arbitrary. Denote the inverse of $B^{(i)}$ by $D^{(i)}$. Basic solutions are defined by $x_B^{(i)} = D^{(i)}\beta^{(i)}$.

Let us apply the following transformation of the system of coordinates:

$$R^n \ni x \mapsto \bar{x} = B^{(i)}x \in R^n \tag{7.1}$$

to LLP. Row vectors are transformed as linear functionals on R^n,

$$R_n \ni x \mapsto \bar{x} = xD^{(i)} \in R_n. \tag{7.2}$$

We obtain $ax = aD^{(i)}B^{(i)}x = y^{(i)}\bar{x}$ for the cost function and $A_jx = A_jD^{(i)}B^{(i)}x = y_j^{(i)}\bar{x}$ for the left-hand side of the j-th constraint. The transformed problem thus has the form:

Minimize $\bar{x} \mapsto y^{(i)}\bar{x}$ subject to $\bar{x} \in \bar{w} = B^{(i)}w$, where the admissible set $\bar{w}$ is determined by

$$y_l^{(i)}\bar{x} \leqslant b_l, \ l \in c_I, \qquad y_l^{(i)}\bar{x} = b_l, \ l \in c_E.$$

Similarly as in simplex, the vectors $y^{(i)}$ and $y_l^{(i)}, l \in c_I \cup c_E$, can be calculated by means of recursive formulae, which considerably reduces computational effort. For details see Lemma 2 where recursive formulae are also given for the inverse of the basic matrix $D^{(i)}$ and the basic solution $x_B^{(i)}$. It should be remembered that these formulae are numerically unstable, and so, for large problems, matrix factorization techniques analoguous to those of revised simplex are recommended [7].

The reason for the use of variable system of coordinates, determined in each iteration by the current basic matrix $B^{(i)}$, is that in this system the computations both of Part A and Part B take a much simpler form. This results from the fact that $y_l^{(i)}, l \in E_B^{(i)}$, are unit vectors. In consequence, we have, for example, $\bar{x}_B^{(i)} = B^{(i)}x_B^{(i)} = \beta^{(i)}$. The simplification is most clearly seen in the typical case where $E^{(i)} = E_B^{(i)}$. As has been pointed out, in iterations of Part A it is essential to have an easy way of determining a search direction

$$d^{(i)} \in K_0 \cap K_E^{(i)} \quad \text{or} \quad \bar{d}^{(i)} \in \bar{K}_0^{(i)} \cap \bar{K}_E^{(i)},$$

where $\bar{d}^{(i)} = B^{(i)}d^{(i)}$, $\bar{K}_0^{(i)} = B^{(i)}K_0$ and $\bar{K}_E^{(i)} = B^{(i)}K_E^{(i)}$. The variable system of coordinates offers such a possibility: $\bar{d} \in \bar{K}_0^{(i)} \cap \bar{K}_E^{(i)}$ if and only if

$$\bar{d}_{u(l)} \leqslant 0 \ \forall l \in E^{(i)} \cap c_I, \quad \bar{d}_{u(l)} = 0 \ \forall l \in E^{(i)} \cap c_E, \quad \text{and} \quad y^{(i)}\bar{d} < 0.$$

This is simpler than the corresponding formulation in the original system of coordinates, as can be seen in (3.1). Let us mention that a contrary effect is observed for some relationships. For example, the simplest rule for starting points $v^{(i+1)} = x^{(i)}$ corresponds to

$$\bar{v}^{(i+1)} = B^{(i+1)}D^{(i)}\bar{x}^{(i)}$$

in the variable system of coordinates.

7.2. *Elimination of equality constraints.* The problem LP can be reduced to a problem with inequality constraints only by means of the Gauss elimination of variables applied to system (1.1). A similar effect is achieved if $C_E \subset E^{(i)}$ for every i in all subproblems LLP(k) or, equivalently, $C_E \subset E^{(1)}$ for $k = 1$. The former method reduces the dimension of the linear programming problem and allows us to use a simplified form of the algorithm but requires a considerable amount of computations for the elimination and for the transformation of the final solution back to the original system of coordinates. In the latter method we avoid these, but in every iteration deal with the vector of variables of full dimension.

7.3. *Determination of $d^{(i)}$, $v^{(i)}$ and $E^{(i)}$ in Part A.* In the general framework of inner search methods there is a considerable freedom of choice for $d^{(i)}$, $E^{(i)}$ and $v^{(i)}$. Restrictions only result from the general features of methods of admissible directions and conditions of finite convergence. Below we give a few more detailed rules which have already been tested to an extent.

Typically, in Part A, $v^{(i)} = x^{(i-1)}$, $E^{(i)}$ has n elements and $E_B^{(i)} = E^{(i)}$. Moreover, $E^{(i)}$ consists of constraints which were active in one of the previous iterations. An exchange of $E^{(i)}$ occurs if $A_j x^{(i)} = b_j$ for some $j \in E^{(i)}$; j then enters $E^{(i+1)}$ in place of the oldest element of $E^{(i)}$ such that $E^{(i+1)}$ is independent. To avoid zigzagging, we may put $E^{(i+1)} = E^{(i)} \cup \{j\}$ if A_j is linearly dependent on $t < n$ vectors A_l, $l \in E^{(i)}$, or is close to it, and j was active in some previous iterations. Alternatively, the checking algorithm is then activated. As an example, consider a two-sided constraint $b_j^- \leqslant A_j x \leqslant b_j^+$, $b_j^- < b_j^+$. If the upper and lower constraints are alternately active in several successive iterations, then both enter the current set $E^{(i)}$ and remain there for a fixed number of iterations. The starting point $v^{(i)}$ is then shifted into the relative interior of the admissible set.

Search directions $d^{(i)}$ in Part A are linear combinations of a vector γ pointing into the relative interior of the cone $K_E^{(i)}$ and a descent direction η,

$$d^{(i)} = \lambda\gamma + (1-\lambda)\eta, \quad \lambda \geqslant 0.$$

Below, vectors with bars have the meaning as in (7.1) and (7.2). For example, we can take

$$\bar{\gamma} = (-1, -1, \ldots, -1)^T \quad \text{if } E^{(i)} = E_B^{(i)}.$$

This choice for $\bar{\gamma}$ gives the best results if the vectors A_l, $l \in E^{(i)}$, have approximately equal norms. For η we can take $-a^T$ (then $\bar{\eta} = -B^{(i)}B^{(i)^T}y^{(i)^T}$) or, in the variable system of coordinates, a vector $\bar{\eta}$, $\bar{\eta}_i = -\delta_{ip}$ for every i, where δ_{ip} is the Kronecker symbol and p is determined by

$$[y^{(i)}]^p = \max_l [y^{(i)}]^l,$$

$[y^{(i)}]^l$ is the l-th element of $y^{(i)}$. The coefficient λ may be determined by means of a rough optimization.

7.4. *Starting points in Part B.* The algorithm of Part B has a better rate of convergence if the starting points $v^{(i)}$ are situated in w° not far from the axis connecting the optimal solution with the centre of w. In testing computations a constant starting point was used, $v^{(i)} = \text{const}$, constructed as a linear combination of the points $x^{(i)}$ obtained in the preceding algorithm of Part A.

8. Conclusions. The conditions of finiteness of algorithms of Part A (Section 4) are the main theoretical result of this paper. The theorems on finite convergence of algorithms of Part A and Part B (Sections 3–5) create a framework for the construction of various practical inner search methods. A few suggestions in this direction have been given in Section 7. Although the computational experience with inner search is fragmentary by now, the results show that it works better than simplex in linear problems in which the search paths of simplex have many vertices per unit of length, especially if the difference $V-n$ is large, where V is the number of vertices. This is normal if the number of inequality constraints M is much greater than n. Simplex methods prevail for large n and small $M-n$, especially when the size of the admissible set varies by many orders of magnitude along different axes of any smallest basis which spans the admissible set. A thorough comparison of inner search with simplex and related methods requires more computational experience and will be the subject of a future work.

References

[1] R. Gabasov and F. M. Kirillova, *Methods of Linear Programming* (in Russian), BGU, Minsk, Part 1 – 1977, Part 2 – 1978, Part 3 – 1980.

[2] — and A. I. Tyatyushkin, *Constructive Methods of Optimization*, Part 1 – *Linear Problems* (in Russian), Universitetskoe, Minsk 1984.

[3] M. K. Gavurin and V. N. Malozemov, *Extremal Problems with Linear Constraints* (in Russian), University of Leningrad, 1984.

[4] M. Kallio and W. Orchard-Hays, *Experiments with the reduced gradient method for general and dynamic linear programming*, pp. 631–662 in: G. B. Dantzig, M. A. H. Dempster and M. J. Kallio (eds.), *Large-Scale Linear Programming*, Vol. 2, IIASA, Laxenburg 1981.

[5] M. Kallio and E. Porteus, *A class of methods for linear programming*, Math. Programming 14 (1978), pp. 161–169.

[6] A. Korytowski, *A class of methods of admissible directions for linear programming*, pp.

1850–1851 in: *Proc. 25th IEEE Conference on Decision and Control*, Vol. 3, Athens, December 10–12, 1986.

[7] B. A. Murtagh, *Advanced Linear Programming: Computation and Practice*, McGraw-Hill, New York 1981.

INSTITUTE OF AUTOMATICS
ACADEMY OF MINING AND METALLURGY
AL. A. MICKIEWICZA 30, 30-059 KRAKÓW, POLAND

Received on 1987.04.27;
revised version on 1987.10.05