ON SEQUELS OF THE SUPPORTING HYPERBALLS METHOD BY F. NOŽIČKA

HORST HOLLATZ
Otto–von–Guericke–University Magdeburg, Germany
http://horst.hollatz.de

Abstract

In 1964, F. Nožička published his supporting hyperballs method for solving linear programming problems. We present finite algorithmic models containing this method as a particular case. The main tool is here an addition to the well-known Farkas Lemma that restricts the selection of feasible descent directions in a boundary point of the feasible domain to a linear subspace. With the aid of projection and reduction methods, an extension to quadratic problems is possible. The algorithmic models are constructed in such a way that finiteness without constraint qualifications is guaranteed, and, in addition, numerically stable implementations are possible. Numerical tests for problems on which the simplex method attains exponential complexity confirm the high (evidently, dimension-independent) efficiency of the models.

Keywords: linear programming, quadratic programming, linear inequalities

Basic for the sequel is the following assertion:
Given are an $m \times n-$matrix $\mathsf{A} \in \mathbb{R}^{m\times n}$ and a vector $\mathsf{c} \in \mathbb{R}^n$. If the linear inequality system

$$\mathsf{Ay} \leqq \mathsf{o}, \quad \mathsf{c}^T\mathsf{y} < 0$$

is solvable then it also possesses a solution of the form

$$\mathsf{y} = -(u_0\mathsf{c} + \mathsf{A}^T\mathsf{u}),\ u_0 \geqq 0, \mathsf{u} \geqq \mathsf{o}.$$

Proof. The solution set of the inequality system and the column space of the matrix $[\mathsf{A}^T, \mathsf{c}]$ are convex; thus the separation theorem for convex sets is applicable. We suppose that both sets are disjoint. Then, according to the separation theorem [1] there is a hyperplane that separates both sets. Thus, there exists a non-zero vector $\mathsf{d} \in \mathbb{R}^n$ such that

$$\mathsf{d}^T\mathsf{x} \leqq 0 \quad \forall \mathsf{x} :\ \mathsf{x} = -(u_0\mathsf{c} + \mathsf{A}^T\mathsf{u}),\ u_0 \geqq 0, \mathsf{u} \geqq \mathsf{o}.$$

and

$$\mathsf{d}^T\mathsf{x} \geqq 0 \quad \forall \mathsf{x} :\ \mathsf{Ax} \leqq \mathsf{o}, \mathsf{c}^T\mathsf{x} < 0.$$

It follows from the last condition that, in accordance with the Farkas Lemma, the system

$$\mathsf{d} = -(\mathsf{A}^T\mathsf{u} + u_0\mathsf{c}),\ \ u_0 \geqq 0, \mathsf{u} \geqq \mathsf{o}$$

is solvable. Inserting this into the first condition, we get $\mathsf{d}^T\mathsf{d} \leqq 0$ or $\mathsf{d} = \mathsf{o}$. This contradiction proves our assertion. q.e.d.
Since both systems $\mathsf{A}\mathsf{y} \leqq \mathsf{o}, \mathsf{c}^T\mathsf{y} < 0$ and $\mathsf{y} = u_0\mathsf{c} + \mathsf{A}^T\mathsf{u}$ are homogeneous, we can replace the first one by $\mathsf{A}\mathsf{y} \leqq \mathsf{o}, \mathsf{c}^T\mathsf{y} = -1$.
In [4] we substantiated an algorithmic model for linear programming problems with $\varepsilon-$active index sets. Here we consider the case $\varepsilon = 0$ and extend it to quadratic problems. Firstly we consider the linear programming problem

$$\mathcal{L}: \quad \min\{\, \mathsf{c}^T\mathsf{x} \mid \mathsf{A}\mathsf{x} \leqq \mathsf{b}\,\}$$

with $\mathsf{c} \in \mathbb{R}^n, \mathsf{b} \in \mathbb{R}^m, \mathsf{A} = (a_{ij})_{m,n}$. Let $\mathsf{A}_1^T, \ldots, \mathsf{A}_m^T$ be the rows of the matrix A. For each vector x we denote by A_x the submatrix of A consisting exactly of the rows A_i satisfying $\mathsf{A}_i^T\mathsf{x} = b_i$.
We assign to each feasible point x of problem $\mathcal{L}$ the set

$$\Upsilon(\mathsf{x}) = \{\, \mathsf{y} \mid \mathsf{A}_\mathsf{x}\mathsf{y} \leqq \mathsf{o}, \mathsf{c}^T\mathsf{y} = -1, \mathsf{y} = u_0\mathsf{c} + \mathsf{A}_\mathsf{x}^T\mathsf{u}\,\}.$$

The dimension of the vector u is chosen in such a way that the equalities make sense. Thus, we have in this case $\mathsf{u} \in \mathbb{R}^{|I(\mathsf{x})|}$ where $I(\mathsf{x})$ is the active index set for a feasible point x:

$$I(\mathsf{x}) = \{\, i \mid \mathsf{A}_i^T\mathsf{x} = b_i\,\}.$$

If the set $\Upsilon(\mathsf{x})$ is non-empty, then all vectors from this set are descent directions for the objective function. Therefore we call the set $\Upsilon(\mathsf{x})$ **feasible descent polyhedron** in the point x. For optimal solutions x^* (for them only) the feasible descent polyhedron is empty since we have

$$-\mathsf{c} = \mathsf{A}_{\mathsf{x}^*}^T\mathsf{u}, \quad \mathsf{u} \geqq \mathsf{o}$$

and, according to the assertion formulated at the beginning, the system

$$\mathsf{A}_{\mathsf{x}^*}\mathsf{y} \leqq \mathsf{o}, \quad \mathsf{c}^T\mathsf{y} = -1$$

has no solutions of the form $\mathsf{y} = u_0\mathsf{c} + \mathsf{A}_{\mathsf{x}^*}^T\mathsf{u}$.
The connection with the supporting hyperballs method by F. Nožička [3] is clarified by the following considerations. Without loss of generality, we can assume that the normal vectors A_i, c of the hyperplanes

$$H_i = \{\, \mathsf{x} \mid \mathsf{A}_i^T\mathsf{x} = 0\,\},\ i = 1, \ldots, m,\ H_0 = \{\, \mathsf{x} \mid \mathsf{c}^T\mathsf{x} = 0\,\}$$

are normalized: $\|\mathsf{A}_i\| = 1, i = 1, \ldots, m,\ \|\mathsf{c}\| = 1$.
If we select, as the feasible direction polyhedron in a vertex of the regular feasible set, the set

$$\{\, \mathsf{y} \mid \mathsf{A}_\mathsf{x}\mathsf{y} \leqq -1\,\}$$

and its extreme point y as the solution of $\mathsf{A}_\mathsf{x}\mathsf{y} = -1$, then all points of ray

$$\mathsf{x}(\lambda) = \mathsf{x} + \lambda\mathsf{y}, \lambda > 0$$

have distance λ from all active boundary hyperplanes. This ray is the axis introduced by Nožička. The ball K_λ with center $\mathsf{x}(\lambda)$ and radius λ touches all active hyperplanes. If the point $\mathsf{x}^*(\lambda)$ is the minimizer of the objective function on the ball K_λ, then $\mathsf{x}^*(\lambda), \lambda > 0$ represents a ray and lies entirely in the feasible descent polyhedron defined above. With the additional assumption that the axis is orthogonal to the boundary face Nožička managed to define an axis in each boundary point ([3]).
The idea of the algorithmic model to be justified here is the following.
Step 1 (Minimization on a closed boundary face).
Starting with a feasible solution x^0 of problem $\mathcal{L}$ we minimize the objective function over the closed boundary face $\overline{\mathcal{S}}(\mathsf{x}^0)$ of the feasible domain defined by $I^0 = I(\mathsf{x}^0)$. Let $\mathsf{x}^*(I^0)$ be the corresponding optimal solution. If this solution is optimal for problem $\mathcal{L}$, then the method terminates.
Step 2 (Transition to another face).
We determine a descent direction $\mathsf{y}^*(I^0)$ in the point $\mathsf{x}^*(I^0)$ directed towards the (relative) interior of the feasible domain. Choosing an optimal step length along this direction we obtain a new feasible solution x^1 on a boundary face characterized by the active index set $I(\mathsf{x}^1)$. We go back to Step 1.
Many well-known methods use in Step 2 an inactivating strategy based on the values of the Lagrange multipliers (cf. [6] and the literature cited therein). The inactivation of a constraint is undertaken in such a way that multiple inactivation can lead to infeasible directions. Here we avoid this by restricting the set of feasible directions. For the minimization over a boundary face of the feasible domain, we need the set $\Upsilon^0(\mathsf{x})$ of all extreme points of the descent polyhedron:

$$\Upsilon^0(\mathsf{x}) = \{\, \mathsf{y} \mid \mathsf{A}_\mathsf{x}\mathsf{y} = \mathsf{o}, \mathsf{c}^T\mathsf{y} = -1, \mathsf{y} = u_0\mathsf{c} + \mathsf{A}_\mathsf{x}^T\mathsf{u} \,\}.$$

The extreme point can be computed in a numerically stable way. Let the matrix $[\mathsf{A}_\mathsf{x}^T, \mathsf{c}]$ be column regular. Then there is exactly one extreme point. We perform a QR–factorization of the matrix $[\mathsf{A}_\mathsf{x}^T, \mathsf{c}]$:

$$\mathsf{Q}^T[\mathsf{A}_\mathsf{x}^T, \mathsf{c}] = \begin{bmatrix} \mathsf{R} \\ \mathsf{O} \end{bmatrix}.$$

Here $\mathsf{R} = (r_{ij})_{p,p}$ is an upper triangular matrix of order $p = |I(x)| + 1$. Then we have

$$\mathsf{Q}^T\mathsf{y} = \begin{bmatrix} \mathsf{R} \\ \mathsf{O} \end{bmatrix} \begin{bmatrix} \mathsf{u} \\ u_0 \end{bmatrix} = \begin{bmatrix} \mathsf{z} \\ \mathsf{o} \end{bmatrix}$$

where $\mathsf{z} \in \mathbb{R}^p$. Thus

$$\mathsf{y} = \mathsf{Q}\begin{bmatrix} \mathsf{z} \\ \mathsf{o} \end{bmatrix}.$$

The condition

$$\begin{bmatrix} \mathsf{A}_\mathsf{x} \\ \mathsf{c}^T \end{bmatrix} \mathsf{y} = -\mathsf{e}_p$$

implies that, after substituting the factorization, the unknown vector z can be determined as the solution of the system

$$\mathsf{R}^T\mathsf{z} = -\mathsf{e}_p.$$

This yields the solution

$$\mathsf{z} = -\frac{1}{r_{pp}}\mathsf{e}_p.$$

The extreme point of the descent polyhedron is then

$$\mathsf{y} = -\frac{1}{r_{pp}}\mathsf{Q}\mathsf{e}_p.$$

In this representation we use the same symbol e_p for the $p-$th unit vector in each vector space. The Lagrange multipliers are obtained by solving the system

$$\mathsf{R}\begin{bmatrix} \mathsf{u} \\ u_0 \end{bmatrix} = -\frac{1}{r_{pp}}\mathsf{e}_p.$$

In the case $u_0 > 0, \mathsf{u} \geqq \mathsf{o}$ the current solution is optimal.
For the descent polyhedron we have the description

$$\Upsilon(\mathsf{x}) = \{\, \mathsf{y} \mid \mathsf{y} = \mathsf{Q}\begin{bmatrix} \mathsf{z} \\ \mathsf{o} \end{bmatrix},\ \mathsf{R}^T\mathsf{z} \leqq -\mathsf{e}_p \,\}.$$

In a vertex x with regular matrix A_x, the QR$-$factorization of the matrix $[\mathsf{A}_\mathsf{x}^T, \mathsf{c}]$ has the form

$$[\mathsf{A}_\mathsf{x}^T, \mathsf{c}] = \mathsf{Q}[\mathsf{R}, \bar{\mathsf{c}}].$$

Therefore the descent polyhedron is

$$\Upsilon(\mathsf{x}) = \{\, \mathsf{y} \mid \mathsf{y} = \mathsf{Q}\mathsf{z},\ \mathsf{R}^T\mathsf{z} \leqq \mathsf{o},\ \bar{\mathsf{c}}^T\mathsf{z} = -1 \,\}.$$

The Lagrange multipliers vector u, related to a vertex, is the solution of the system

$$\mathsf{R}\mathsf{u} = -\bar{\mathsf{c}}.$$

In the case $\mathsf{u} \geqq \mathsf{o}$ the vertex is optimal; otherwise there is the following possibility for finding a feasible descent direction. Let $\mathsf{v} \in \mathbb{R}^n$ with $\mathsf{v} \geqq \mathsf{o}, \mathsf{u}^T\mathsf{v} = -1$ and $\bar{\mathsf{z}}$ be the solution of $\mathsf{R}^T\mathsf{z} = -\mathsf{v}$. Then we have

$$\bar{\mathsf{c}}^T\bar{\mathsf{z}} = -\mathsf{u}^T\mathsf{R}^T\bar{\mathsf{z}} = \mathsf{u}^T\mathsf{v} = -1,$$

and the vector $\mathsf{y} = \mathsf{Q}\bar{\mathsf{z}}$ is a feasible descent direction, i. e. $\mathsf{y} \in \Upsilon(\mathsf{x})$.
The transition to a new feasible solution changes the matrix $[\mathsf{A}_\mathsf{x}^T, \mathsf{c}]$ by adding or deleting some columns. This change can be realized by updating the current QR$-$factorization [6].
Let x be a feasible solution to problem $\mathcal{L}$ and $\Upsilon^0(\mathsf{x}) \neq \emptyset$. We choose $\mathsf{y} \in \Upsilon^0(\mathsf{x})$. For the ray $\mathsf{x}(\lambda) = \mathsf{x} + \lambda\mathsf{y}$ we get then

$$\mathsf{c}^T\mathsf{x}(\lambda) = \mathsf{c}^T\mathsf{x} - \lambda,$$

and for $\lambda > 0$ we should take the value for which the ray leaves the feasible domain:

$$\lambda = \Lambda(\mathsf{x}, \mathsf{y}) = \min\left\{ \frac{b_i - \mathsf{A}_i^T\mathsf{x}}{\mathsf{A}_i^T\mathsf{y}} \mid \mathsf{A}_i^T\mathsf{y} > 0,\ i \notin I(\mathsf{x}) \right\}.$$

If the descent polyhedron contains no extreme points then each point of the corresponding boundary face is a minimal point of the objective function with respect to this face. Summarizing, we can represent the minimization of the objective function over a boundary face in the form

$$\textbf{while } (\mathsf{y} = \Phi^0(\Upsilon^0(\mathsf{x}))) \neq \mathsf{o} \textbf{ do } \mathsf{x} := \mathsf{x} + \Lambda(\mathsf{x}, \mathsf{y})\mathsf{y}.$$

The iteration starts with an arbitrary feasible point; Φ^0 is a selection function for the corresponding set ($\Phi^0(\emptyset) = \mathsf{o}$). In every iterative step we add at least one constraint to the set of active constraints. Thus the iteration ends with $\Upsilon^0(\mathsf{x}) = \emptyset$ and therefore with a feasible solution $\bar{\mathsf{x}} = \mathsf{x}^*(\mathsf{x})$ minimizing the objective function over the closed boundary face $\overline{\mathcal{S}}(\mathsf{x})$. If the descent polyhedron at the point $\bar{\mathsf{x}}$ is empty (that is $\Upsilon(\bar{\mathsf{x}}) = \emptyset$) then the point $\bar{\mathsf{x}}$ is an optimal solution to the problem $\mathcal{L}$. This test is equivalent to the optimality test for the point $\bar{\mathsf{x}}$, and here it is realized by computing the Lagrange multipliers.
Otherwise we select a vector $\bar{\mathsf{y}} \in \Upsilon(\bar{\mathsf{x}})$ and set

$$\mathsf{x} = \bar{\mathsf{x}} + \Lambda(\bar{\mathsf{x}}, \bar{\mathsf{y}})\bar{\mathsf{y}},$$

which yields the objective function value

$$\mathsf{c}^T\bar{\mathsf{x}} - \Lambda(\bar{\mathsf{x}}, \bar{\mathsf{y}}).$$

The selection of $\bar{\mathsf{y}} \in \Upsilon(\bar{\mathsf{x}})$ can be done in different ways (see above). Summing up, we obtain an algorithmic model starting with an arbitrary feasible solution:

$$\mathcal{A}(\mathcal{L}, \mathsf{x}): \quad \begin{cases} \textbf{while } (\bar{\mathsf{y}} = \Phi(\Upsilon(\mathsf{x}))) \neq \mathsf{o} \textbf{ do} \\ \{ \textbf{ while } (\mathsf{y} = \Phi^0(\Upsilon^0(\mathsf{x}))) \neq \mathsf{o} \textbf{ do } \mathsf{x} := \mathsf{x} + \Lambda(\mathsf{x}, \mathsf{y})\mathsf{y} \\ \quad \mathsf{x} := \mathsf{x} + \Lambda(\mathsf{x}, \bar{\mathsf{y}})\bar{\mathsf{y}} \ \}. \end{cases}$$

Here Φ^0 and Φ are selection functions for the respective sets.

Theorem 1 *Let a solvable problem*

$$\min\{ \mathsf{c}^T\mathsf{x} \mid \mathsf{A}\mathsf{x} \leqq \mathsf{b} \}$$

be given. Then, for an arbitrary feasible starting point x*, the algorithmic model* $\mathcal{A}(\mathcal{L}, \mathsf{x})$ *yields an optimal solution in a finite number of steps.*

Proof. Having in mind the description above, we note that the rules for minimizing the objective function over a boundary face of the feasible domain imply the termination of the process after at most n steps. If the corresponding point is not optimal, we go to a new boundary face where the objective function takes a smaller value. Therefore, no cycles of boundary faces can arise in this algorithmic model. q.e.d.
It is interesting to note that the algorithmic model does not require a constraint qualification; however, for a particular implementation of the iterative steps, we are compelled to take into account a constraint qualification for the problem in question. We illustrate the performance of this scheme on a numerical example. Consider the problem (with $\varepsilon \in (0, 0.5)$)

$$\min\{ -\mathsf{e}_n^T\mathsf{x} \mid 0 \leqq x_1 \leqq 1,\ \varepsilon x_{i-1} \leqq x_i \leqq 1 - \varepsilon x_{i-1},\ i = 2, \ldots, n \}.$$

This is a typical problem for which the simplex method with the starting point $\mathsf{x} = \mathsf{o}$ enumerates all 2^n vertices of the feasible domain [5] before finding the obvious optimal solution $\mathsf{x}^* = \mathsf{e}_n$. We implemented an algorithm where the vector v (see above) is chosen according to the following rule. Let u be the n-dimensional Lagrange multiplier vector in a vertex. For $q = |\{ i \mid u_i < 0 \}|$ we set

$$qv_i = \begin{cases} -\dfrac{1}{u_i}, & u_i < 0 \\ 0, & \text{otherwise} \end{cases} .$$

The following table represents the number of iterations for various values of ε and n.

	10	20	30	50	100	200	500
0.05	9	9	9	9	9	9	9
0.10	11	12	12	12	12	12	12
0.15	11	14	14	14	14	14	14
0.20	11	16	16	16	16	16	16
0.25	11	18	18	18	18	18	18
0.30	11	21	21	21	21	21	21
0.35	12	22	24	24	24	24	24
0.40	12	22	53	53	53	53	53
0.45	12	22	31	31	31	31	31

If we choose the vector v as

$$\mathsf{v} = -\frac{1}{\|\mathsf{u}_-\|^2}\mathsf{u}_-,$$

where the vector u_- contains only the negative Lagrange multipliers (other components are zeroes), we get the following table:

	10	20	30	50	100	200	500
0.05	8	8	8	8	8	8	8
0.10	10	10	10	10	10	10	10
0.15	10	13	13	13	13	13	13
0.20	10	15	15	15	15	15	15
0.25	10	17	17	17	17	17	17
0.30	10	19	19	19	19	19	19
0.35	10	20	22	22	22	22	22
0.40	10	20	25	25	25	25	25
0.45	10	20	29	29	29	29	29

Note that the number of iterations depends on the criterion for including a constraint into the active constraint set. In the above mentioned cases we took $b_i - \mathsf{A}_i^T\mathsf{x} < 10^{-10}$. If we take $b_i - \mathsf{A}_i^T\mathsf{x} < 10^{-2}$ then the number of iterations will be reduced drastically. For the first case we have the following table.

	10	20	30	50	100	200	500
0.05	3	3	3	3	3	3	3
0.10	4	4	4	4	4	4	4
0.15	4	4	4	4	4	4	4
0.20	4	4	4	4	4	4	4
0.25	5	5	5	5	5	5	5
0.30	5	5	5	5	5	5	5
0.35	7	7	7	7	7	7	7
0.40	13	13	13	13	13	13	13
0.45	8	8	8	8	8	8	8

If we choose the vector v in the form

$$\mathsf{v} = -\frac{1}{u_r}\mathsf{e}_r$$

(with $u_r = \min\{\, u_i \mid u_i < 0 \,\}$) then we obtain the simplex method.
We consider now the quadratic problem

$$\mathcal{Q}: \quad \min\{\, f(\mathsf{x}) \mid \mathsf{A}\mathsf{x} \leqq \mathsf{b} \,\}$$

with $f(\mathsf{x}) = \mathsf{c}^T\mathsf{x} + \frac{1}{2}\mathsf{x}^T\mathsf{C}\mathsf{x}$ and a symmetric positive definite $n \times n-$matrix C. In this case, for a given feasible point x, we can reduce the minimization of the objective function over the affine subspace $\{\, \mathsf{x} \mid \mathsf{A}_i^T\mathsf{x} = b_i,\ i \in I \,\}$ defined by the active index set $I = I(\mathsf{x})$, to the solution of the following linear equation system

$$\begin{bmatrix} \mathsf{C} & \mathsf{A}_\mathsf{x}^T \\ \mathsf{A}_\mathsf{x} & 0 \end{bmatrix} \begin{bmatrix} \mathsf{y} \\ \mathsf{u} \end{bmatrix} = \begin{bmatrix} f'(\mathsf{x}) \\ \mathsf{o} \end{bmatrix}.$$

If the matrix A_x is row regular, and y is the solution to this system then the point $\mathsf{x}+\mathsf{y}$ is the minimizer on the affine subspace [6]. In the case $\mathsf{y} = \mathsf{o}$ the vector x is readily the minimizer. Let Ψ be a minimizing operator over a subspace:

$$\mathsf{y} = \Psi(f', \mathsf{x}).$$

The operator Ψ, similarly to the previous case, can be implemented (in a numerically stable way) by means of a QR$-$factorization of the matrix A_x^T and a Cholesky-factorization of the matrix C. The determination of the step size has to be modified as follows

$$\lambda = \min\{\, 1, \Lambda(\mathsf{x}, \mathsf{y}) \,\}.$$

The descent polyhedron is now

$$\Upsilon(\mathsf{x}) = \{\, \mathsf{y} \mid \mathsf{A}_\mathsf{x}\mathsf{y} \leqq \mathsf{o}, f'(\mathsf{x})^T\mathsf{y} = -1, \mathsf{y} = u_0 f'(\mathsf{x}) + \mathsf{A}_\mathsf{x}^T\mathsf{u} \,\}.$$

In the case $\Upsilon(\mathsf{x}) = \emptyset$ we have readily the solution. If y is a feasible descent direction then the minimum of the function f over the ray $\mathsf{x} + \lambda\mathsf{y}$ is attained at the point

$$\mathsf{x} + \frac{1}{\mathsf{y}^T\mathsf{C}\mathsf{y}}\mathsf{y}.$$

Thus we have the following algorithmic model:

$$\mathcal{A}(\mathcal{Q}, \mathsf{x}): \quad \left\{ \begin{array}{l} \textbf{while } (\bar{\mathsf{y}} = \Phi(\Upsilon(\mathsf{x}))) \neq \mathsf{o} \textbf{ do} \\ \{\, \textbf{while } (\mathsf{y} = \Psi(f', \mathsf{x})) \neq \mathsf{o} \textbf{ do } \mathsf{x} := \mathsf{x} + \min\{\, 1, \Lambda(\mathsf{x}, \mathsf{y}) \,\}\mathsf{y} \\ \quad \mathsf{x} := \mathsf{x} + \min\{\, \frac{1}{\bar{\mathsf{y}}^T\mathsf{C}\bar{\mathsf{y}}}, \Lambda(\mathsf{x}, \bar{\mathsf{y}}) \,\}\bar{\mathsf{y}} \,\}. \end{array} \right.$$

The finiteness of this scheme can be proved similarly to the linear case.

Theorem 2 *For an arbitrary feasible starting point* x*, the algorithmic model* $\mathcal{A}(\mathcal{Q}, \mathsf{x})$ *solves the quadratic problem* $\mathcal{Q}$ *in a finite number of steps.*

As an illustrative example we set $\mathsf{C} = \mathsf{E}$ and modify thereby the objective function of the previous example. We obtain the following table (for the case when the active index set is determined by $b_i - \mathsf{A}_i^T \mathsf{x} < 10^{-10}$).

	10	20	30	50	100	200	500
0.05	7	7	7	7	7	7	7
0.10	7	10	10	10	10	10	10
0.15	4	8	8	8	8	8	8
0.20	4	7	7	7	7	7	7
0.25	2	9	9	9	9	9	9
0.30	2	8	12	12	12	12	12
0.35	2	6	10	10	10	10	10
0.40	2	4	13	13	13	13	13
0.45	2	4	13	13	13	13	13

References

[1] Stoer, J., and Ch. Witzgall:
Convexity and Optimization in Finite Dimensions I.
Springer-Verlag, Berlin, Heidelberg, New York 1970

[2] Tschernikow, S. N.: Lineare Ungleichungen.
Deutscher Verlag der Wissenschaften, Berlin 1971

[3] Nožička, F.: Über eine Berührungshyperkugel-Methode in der Theorie der Programmierung.
Mathematik und Kybernetik in der Ökonomie,
Tagungsprotokoll, Teil II, 26-53, Berlin 1964

[4] Hollatz, H: Eine Alternativ-Methode für die numerische Lösung linearer Optimierungsaufgaben.
in: Operations Research Proceedings 1996, ed. U. Zimmermann et al., Springer-Verlag Berlin, Heidelberg, New York 1997, 5-9

[5] Klee V., G. Minty: How Good is the Simplex-Algorithm?
in: Inequalities III, ed. O. Shisha, Akademic Press, New York 1972, 159-175

[6] Spellucci, P.: Numerische Verfahren der nichtlinearen Optimierung.
Birkhäuser Verlag, Basel, Boston, Berlin 1993