Discussiones Mathematicae
Differential Inclusions, Control and Optimization 22 (2002) 97–123

SELECTION STRATEGIES IN PROJECTION METHODS FOR CONVEX MINIMIZATION PROBLEMS

Andrzej Cegielski and Robert Dylewski

Institute of Mathematics
University of Zielona Góra
ul. Podgórna 50, 65–246 Zielona Góra, Poland

e-mail: a.cegielski@im.uz.zgora.pl
e-mail: r.dylewski@im.uz.zgora.pl

Abstract

We propose new projection method for nonsmooth convex minimization problems. We present some method of subgradient selection, which is based on the so called residual selection model and is a generalization of the so called obtuse cone model. We also present numerical results for some test problems and compare these results with some other convex nonsmooth minimization methods. The numerical results show that the presented selection strategies ensure long steps and lead to an essential acceleration of the convergence of projection methods.

Keywords: convex minimization, projection method, long steps, residual selection, obtuse cone selection.

2000 Mathematics Subject Classification: 65K05, 90C25.

1. Introduction

The aim of the paper is to construct an efficient method for the convex minimization problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in D, \end{array} \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function and $D \subset \mathbb{R}^n$ is a convex and compact subset. The function f being continuous attains its minimum

$$f^* = \min\{f(x) : x \in D\},$$

i.e. the *solution set*

$$M = \operatorname*{Argmin}_{x \in D} f(x) = \{z \in D : \forall_{x \in D} \quad f(z) \leq f(x)\}$$

is nonempty. A general model of such methods was presented in [8] and in [2]. The method has the general form

$$\begin{aligned} x_1 &\in D \quad \text{arbitrary} \\ x_{k+1} &= P_D(x_k + \lambda_k(P_{S(h_k,\alpha_k)}(x_k) - x_k)), \end{aligned} \tag{2}$$

where $\lambda_k \in (0,1)$ is so called *relaxation parameter*, $P_C(x) = \operatorname{argmin}_{z \in C} \|z - x\|$ denotes the metric projection of $x \in \mathbb{R}^n$ onto a closed and convex subset $C \subset \mathbb{R}^n$ with respect to the Euclidean norm $\|\cdot\|$, h_k is a model of the function f (e.g. a polyhedral minorant of f) and $S(h, \alpha)$ denotes a sublevel set of the function $h : \mathbb{R}^n \to \mathbb{R}$ with level α, i.e. $S(h, \alpha) = \{x \in \mathbb{R}^n : h(x) \leq \alpha\}$. We will call such methods the *projection methods with level control* (see also [8]). All the methods under consideration can only use the following information on the problem (1): for any $x \in D$ the objective value $f(x)$ and a single subgradient $g_f(x)$ can be evaluated and for any $x \in \mathbb{R}^n$ the metric projection $P_D(x)$ of x onto D can be evaluated.

The most important parts of any projection method with level control are:

a) construction of a model h_k of the objective function f,
b) updating rules of the level α_k.

Almost all projection methods employ the same updating rules for α_k and have the form $\alpha_k = (1 - \nu)\overline{\alpha}_k + \nu\underline{\alpha}_k$, where $\underline{\alpha}_k$ is a lower bound of the minimal objective value f^* which is updated in each iteration (updating rules depend on the current method) $\overline{\alpha}_k = \min_{1 \leq i \leq k} f(x_i)$ and $\nu \in (0,1)$. The differences in various projection methods consist in various construction of a model h_k. In the simplest method, the variable target value method [7] the function h_k is defined as a linearization f_k of the function f in the point x_k:

$$f_k(x) = g_k^\top(x - x_k) + f(x_k),$$

where

$$x^\top y = \sum_{j=1}^{n} \xi_j \eta_j$$

denotes the standard scalar product of $x = (\xi_1, \ldots, \xi_n)^\top$, $y = (\eta_1, \ldots, \eta_n)^\top \in \mathbb{R}^n$, $g_k = g_f(x_k)$ denotes a subgradient of f at $x_k \in \mathbb{R}^n$. In a variant of bundle methods studied in [12], the model h_k is defined as the best polyhedral minorant of f which can be constructed in the current iteration having all the information obtained in the history of the method, i.e.

$$h_k = \max\{f_i : i = 1, \ldots, k\}.$$

If we denote $f_J = \max\{f_i : i \in J\}$, where $J \subset \{1, \ldots, k\}$, then the above polyhedral model can be written as $f_{\{1,\ldots,k\}}$. Furthermore, in the mentioned variant of bundle methods the lower bound $\underline{\alpha}_k$ has the form $\underline{\alpha}_k = \min\{f_{\{1,\ldots,k\}}(x) : x \in D\}$. In the method of Kim, Ahn and Cho [7] one iteration is extremely cheap. Unfortunately, an extremely slow convergence of the method is observed. On the other hand, in the presented variant of bundle methods of Lemaréchal, Nemirovsky and Nesterov [12] one iteration is extremely expensive, but a good convergence is observed for this method. In this paper, we will seek for a compromise between these two methods. Kiwiel [8] has obtained a surprising result, that both methods are optimal in some sense. One cannot expect that the other method converges essentially better than the first one for all problems. Nevertheless, it seems to be important to find a method which for a wide class of problems converges as well as the presented variant of bundle methods and for which one iteration is relative cheap. We will seek in this paper methods which have this property. The methods constructed in the next part of the paper have the form (2), where $h_k = f_{L_k}$ for some $L_k \subset \{1, \ldots, k\}$. In the simplest case $L_k = \{k\}$ (i.e. only the last iteration is considered in construction of the next approximation x_{k+1}) we obtain the method of Kim, Ahn and Cho [7]. The subset L_k will be constructed such that the evaluation of $P_{S(f_{L_k}, \alpha_k)}(x_k)$ is not too expensive and the next approximation x_{k+1} is "essentially" better than x_k in some sense.

We study in the paper projection methods with level control for the problem (1), of the form

$$\begin{aligned} x_1 &\in D \text{ arbitrary} \\ x_{k+1} &= P_D(x_k + \lambda_k(P_{S_k}(x_k) - x_k)), \end{aligned} \tag{3}$$

where $S_k = S(f_{L_k}, \alpha_k)$ for a model f_{L_k} of f with L_k depending on the method and for a level α_k which is updated in each iteration.

In Section 2, we explain why the choice of $L_k \subset \{1, \ldots, k\}$ which provides long steps $t_k = P_{S_k}(x_k) - x_k$ in the method (3) is advantageous for the good behavior of the method. In Section 3, we present two models of a "cheap" construction of possibly long vectors t_k. In Subsection 3.1, we describe so called obtuse cone model. This model is generalized later in Subsection 3.2. This Subsection contains the main result of the paper. We present so called residual selection model and use this model to a sequential selection of a subset $L_k \subset \{1, \ldots, k\}$ such that relatively long steps $t_k = P_{S(f_{L_k}, \alpha_k)}(x_k) - x_k$ are constructed. Section 4 contains results of numerical tests. We compare the presented selection methods with other known methods for convex nonsmooth minimization: with the bundle trust region method [14] and with the variable target value method [7]. Furthermore, we show the influence of various parameters on the behavior of the residual selection methods.

2. Motivations

Suppose for a moment that the level $\alpha_k > f^*$ in method (3). Then one can show that $M \subset S(f_{L_k}, \alpha_k)$ (see, e.g. [2, Lemma 3]). Furthermore, suppose for the simplicity that the sequence (α_k) is nonincreasing. Then, for $t_i = P_{S_i}(x_i) - x_i$, we have

$$\|x_{k+1} - z\|^2 \leq \|x_1 - z\|^2 - \sum_{i=1}^{k} \lambda_i (2 - \lambda_i) \|t_i\|^2 \tag{4}$$

for all $z \in M$ (see, e.g. [2, Lemma 4]) and, consequently,

$$\sum_{i=1}^{k} \lambda_i (2 - \lambda_i) \|t_i\|^2 \leq \delta^2 \tag{5}$$

for $\delta \geq \mathrm{d}(x_1, M)$, where $\mathrm{d}(x, S) = \inf_{z \in S} \|z - x\|$ denotes the distance of a point $x \in \mathbb{R}^n$ to a subset $S \subset \mathbb{R}^n$. Now we see that it is advantageous to choose $L_k \subset \{1, \ldots, k\}$ providing long vectors t_k. By inequality (4), such a choice leads to a faster convergence of the sequence (x_k) to a solution if $\alpha_k > f^*$. Furthermore, if $\alpha_k \leq f^*$, then for long vectors t_k the inequality converse to (5) can be detected faster which leads to a faster detection that $\alpha_k \leq f^*$.

In this case, the approximation α_k of f^* can be updated (see [7, 10, 2] for details). Now we go to the problem of selection of $L_k \subset \{1, \ldots, k\}$. Since $S_k = S(f_{L_k}, \alpha_k)$ is the solution set to the system of linear inequalities

$$f_i(x) \le \alpha_k,\ i \in L_k,$$

it is convenient for our study to formulate the problem in the following way:

Given a system of linear inequalities

$$G^\top x \le b, \tag{6}$$

where G is a matrix of size $n \times m$, $x \in \mathbb{R}^n$ and $b = (\beta_1, \ldots, \beta_m)^\top \in \mathbb{R}^m$, and an approximation $\overline{x} \in \mathbb{R}^n$ of a solution to this system. Find x^+ which essentially better approximates a solution $x^* \in M_0 = \{x : G^\top x \le b\}$ or detect that $M_0 = \emptyset$.

The best possibility would be to take $x^+ = P_{M_0}\overline{x}$, but the evaluation of such a projection is often too expensive (such a projection was employed in [12] for a variant of bundle methods). On the other hand, x^+ evaluated as a projection $P_{\{x : g_i^\top x \le \beta_i\}}\overline{x}$, where g_i is the i-th column of G and $i \in J = \{1, \ldots, m\}$ is such that $g_i^\top \overline{x} > \beta_i$, is often no essentially better approximation of a solution x^* than $\overline{x}$ and often induces zigzagging behavior which leads to a slow convergence. Note that a variable target value subgradient method of [7] uses such projections. One can also reach a compromise in order to avoid expensive projections and to avoid oscillations: choose appropriate columns $L \subset J$ of the matrix G (denote by G_L the submatrix of G which consists of the chosen columns $L \subset J$ and by b_L the subvector of b which consists of the coordinates $L \subset J$) and evaluate

$$x^+ = P_{\{x : G_L^\top x \le b_L\}}\overline{x}. \tag{7}$$

Now, a new problem arises in this context: how to choose $L \subset J$ such that x^+ evaluated by (7) essentially better approximates a solution x^* than $\overline{x}$ and such that x^+ can easily be evaluated. Suppose that G_L has a full column rank. Then, of course, the equation system $G_L^\top x = b_L$ has a solution. Let $x' = P_{\{x : G_L^\top x = b_L\}}\overline{x}$. It is well known that the Gram matrix $G_L^\top G_L$ of G_L is invertible in this case and that

$$x' = \overline{x} - G_L(G_L^\top G_L)^{-1}(G_L^\top \overline{x} - b_L)\,. \tag{8}$$

Of course, x' is not necessarily equal to x^+ given by (7). Nevertheless, it can easily be shown that

$$x' = x^+ \iff y := (G_L^\top G_L)^{-1}(G_L^\top \overline{x} - b_L) \geq 0 \tag{9}$$

(observe that from inequality in (9) and from equality (8) it follows that

$$\overline{x} - x^+ \in N_{\{x: G_L^\top x \leq b_L\}} x^+$$

and that the last inclusion is equivalent to (7)).

3. Selection models

In this Section, we propose sequential selections of $L \subset J$ such that

$$y = (G_L^\top G_L)^{-1}(G_L^\top \overline{x} - b_L) \geq 0.$$

We present two selection models: the obtuse cone model and the residual selection model.

3.1. Obtuse cone model

Of course, $y \geq 0$ if $G_L^\top \overline{x} \geq b_L$ and $(G_L^\top G_L)^{-1} \geq 0$. This observation leads to a study of properties of a full column rank matrix whose Gram matrix has nonnegative inverse. Various models of selections of $L \subset J$ with such properties were employed for convex feasibility problems or for convex minimization problems in [15, 1, 10, 2]. In these papers, a preselection $J' \subset J$ such that $G_{J'}^\top \overline{x} \geq b_{J'}$ for a current approximation $\overline{x}$ of a solution was made, and then a subset $L \subset J'$ was selected such that G_L has a full column rank and $(G_L^\top G_L)^{-1} \geq 0$. We call such selections as *obtuse cone selections* since the columns of a full column rank matrix A generate (by nonnegative linear combinations) an obtuse cone if and only if $(A^\top A)^{-1} \geq 0$ (see, e.g. [10] or [2, Lemma 8]). The construction of a subsystem G_L generating an obtuse cone and of a new approximation x^+ is based on the Theorem presented below. Various versions of this Theorem can be found in [1, 10, 2, 3]. Let $A = [A_1, a]$ be a matrix of size $n \times m$, where A_1 is a matrix of size $n \times (m-1)$ and $a \in \mathbb{R}^n$ and let $B^+ = (B^\top B)^{-1} B^\top$ denote the Moore-Penrose pseudoinverse of a full column matrix B.

Theorem 1. *Suppose that there exists $y \in \mathbb{R}^n$ such that $A^\top y > 0$. If*

(i) *A_1 has a full column rank*

(ii) $(A_1^\top A_1)^{-1} \geq 0$,

(iii) $A_1^+ a \leq 0$,

then

(iv) *A has a full column rank and*

(v) $(A^\top A)^{-1} \geq 0$.

A sequential application of the above Theorem enables a construction of the mentioned subsystem G_L and of a new approximation x^+. In the next Subsection, we present a more general construction for which the construction of a subsystem G_L generating an obtuse cone is a special case.

Remark 2. If we replace the condition (iii) in Theorem 1 by

(iii′) $A_1^\top a \leq 0$,

then Theorem 1 remains true since

$$A_1^+ a = (A_1^\top A_1)^{-1} A_1^\top a$$

and $A_1^+ a \leq 0$ if $(A_1^\top A_1)^{-1} \geq 0$ and $A_1^\top a \leq 0$. An application of this version of Theorem 1 to a sequential selections of a subset $L \subset J$ and, actually, to a construction of an obtuse cone was employed in [15, 1, 9, 2]. We call such a selection a *regular obtuse cone selection*. The cone generated by the columns of a matrix A obtained by such a selection is called a *regular obtuse cone*.

3.2. Residual selection model

Now we study selections of $L \subset J$ such that $(G_L^\top G_L)^{-1}(G_L^\top \overline{x} - b_L) \geq 0$ without assuming that $G_L^\top \overline{x} - b_L \geq 0$. Such selections lead to longer steps $t = x^+ - \overline{x}$ in comparison with the steps t obtained by obtuse cone selections employed in [15, 1, 9, 2]. To simplify the notation, we denote by A the matrix G_L and by b the vector b_L. Similarly as in the previous Subsection, we write the matrix A of size $n \times m$ in the form $A = [A_1, a]$, where A_1 is a matrix of size $n \times (m-1)$ and $a \in \mathbb{R}^n$. Furthermore, let $b = (b_1^\top, \beta_m)^\top$, where $b_1 = (\beta_1, \ldots, \beta_{m-1})^\top \in \mathbb{R}^{m-1}$ and let $\overline{x} \in \mathbb{R}^n$. Denote by r the *residual vector*, i.e.

$$r = \begin{bmatrix} r_1 \\ \rho \end{bmatrix} = A^\top \overline{x} - b = \begin{bmatrix} A_1^\top \\ a^\top \end{bmatrix} \overline{x} - \begin{bmatrix} b_1 \\ \beta_m \end{bmatrix}, \tag{10}$$

where $r_1 \in \mathbb{R}^{m-1}$ and $\rho \in \mathbb{R}$. The following Theorem renders it possible to construct sequentially a full column rank submatrix A of G for which $y := (A^\top A)^{-1} r \geq 0$. Contrary to the obtuse cone selection model [15, 1, 10, 2] the residual vector r is not necessarily nonnegative. Therefore we will call a model obtained by such a construction a *residual selection model*.

Theorem 3. *Let $\overline{x} \in \mathbb{R}^n$. Suppose that there exists $x' \in \mathbb{R}^n$ such that $A^\top x' < b$. If*

(i) *A_1 has a full column rank,*

(ii) $(A_1^\top A_1)^{-1} r_1 \geq 0$,

(iii) $A_1^+ a \leq 0$,

(iv) $(A_1^+ a)^\top r_1 \leq \rho$,

then

(v) *A has a full column rank,*

(vi) $(A^\top A)^{-1} r \geq 0$.

Proof. a) First we show that there exists $u \in \mathbb{R}^n$ such that $A^\top u > 0$. Let $x^+ = \overline{x} - A_1 (A_1^\top A_1)^{-1} r_1$. Then

$$\begin{aligned} A_1^\top x^+ &= A_1^\top (\overline{x} - A_1 (A_1^\top A_1)^{-1} r_1) \\ &= A_1^\top \overline{x} - r_1 = b_1 \end{aligned}$$

and

$$\begin{aligned} a^\top x^+ &= a^\top (\overline{x} - A_1 (A_1^\top A_1)^{-1} r_1) \\ &= a^\top \overline{x} - a^\top A_1 (A_1^\top A_1)^{-1} r_1 \\ &= a^\top \overline{x} - (A_1^+ a)^\top r_1 \\ &\geq a^\top \overline{x} - \rho = \beta_m, \end{aligned}$$

where the inequality follows from (iv). We see that we have obtained the inequality

$$A^\top x^+ \geq b.$$

Let $x' \in \mathbb{R}^n$ be such that $A^\top x' - b < 0$. We have for $u = x^+ - x'$

$$
\begin{aligned}
A^\top u &= \begin{bmatrix} A_1^\top u \\ a^\top u \end{bmatrix} = \begin{bmatrix} A_1^\top x^+ - A_1^\top x' \\ a^\top x^+ - a^\top x' \end{bmatrix} \\
&= \begin{bmatrix} \overbrace{(A_1^\top x^+ - b_1)}^{=0} - \overbrace{(A_1^\top x' - b_1)}^{<0} \\ \underbrace{(a^\top x^+ - \beta_m)}_{\geq 0} - \underbrace{(a^\top x' - \beta_m)}_{<0} \end{bmatrix} > 0.
\end{aligned} \tag{11}
$$

b) Now we prove that A has a full column rank. Suppose that $\operatorname{rank}(A) < m$. Then $a = A_1 v$ for some $v \in \mathbb{R}^{m-1}$ since A_1 has full column rank. By (iii) we have $v = A_1^+ a \leq 0$. Now we have

$$
0 < a^\top u = v^\top (A_1^\top u) \leq 0. \tag{12}
$$

The contradiction shows that A has a full column rank.

c) Now we show that $(A^\top A)^{-1} r \geq 0$. By [3, Corollary 3.9] we have for $a^\perp = (I - A_1 A_1^+)a$

$$
(A^\top A)^{-1} = \begin{bmatrix} (A_1^\top A_1)^{-1} + \|a^\perp\|^{-2}(A_1^+ a)(A_1^+ a)^\top & -\|a^\perp\|^{-2}(A_1^+ a) \\ -\|a^\perp\|^{-2}(A_1^+ a)^\top & \| a^\perp\|^{-2} \end{bmatrix}
$$

(note that $a^\perp \neq 0$ since A has a full column rank). Consequently,

$$
\begin{aligned}
&(A^\top A)^{-1} r \\
&= \begin{bmatrix} \overbrace{(A_1^\top A_1)^{-1} r_1}^{\geq 0 \text{ by (ii)}} - \| a^\perp\|^{-2} \overbrace{[\rho - (A_1^+ a)^\top r_1]}^{\geq 0 \text{ by (iv)}} \overbrace{A_1^+ a}^{\leq 0 \text{ by (iii)}} \\ \| a^\perp\|^{-2} \underbrace{[\rho - (A_1^+ a)^\top r_1]}_{\geq 0 \text{ by (iv)}} \end{bmatrix} \geq 0.
\end{aligned} \tag{13}
$$

■

A result which slightly differs from the above Theorem can be found in [4].

Remark 4. The assumption $A^\top x' < b$ in Theorem 3 can be weakened. It is enough to assume that $(A^\top x' - b)_m < 0$ or $(A^\top x' - b)_j \cdot (A_1^+ a)_j > 0$ for some $j < m$, where $(y)_i$ denotes the i-th coordinate of a vector $y \in \mathbb{R}^m$. In this case at least one coordinate of the vector $A^\top u$ is positive and at least one of the inequalities in (12) is strict.

Corollary 5. *Let $\overline{x} \in \mathbb{R}^n$ be arbitrary. Suppose that the assumptions of Theorem 3 are satisfied and let*

$$t = -A(A^\top A)^{-1}(A^\top \overline{x} - b). \tag{14}$$

Then

$$x^+ = \overline{x} + t = P_{\{x:A^\top x \le b\}}(\overline{x}). \tag{15}$$

Proof. As we have observed in Section 2, $\overline{x} + t = P_{\{x:A^\top x = b\}}(\overline{x})$. By Theorem 3 and by (9) we obtain equality (15). ∎

In the following Algorithm the properties of the residual selection presented in Theorem 3 and in Corollary 5 are used in order to construct a new approximation x^+ of a solution to the system $G^\top x \le b$ for a current approximation $\overline{x}$. Denote by $r = (\rho_1, \dots, \rho_m)^\top$ the residual vector $G^\top \overline{x} - b$ and by g_i the i-th column of the matrix G, $i \in J = \{1, \dots, m\}$. In order to simplify the notation we suppose that any subset $L = \{j_i, \dots, j_k\} \subset J$ is ordered in agreement with its notation, i.e. j_1 is the first element of L, j_2 is the second element of L and so on. A new element added to a subset L is always the last element of the new subset L', i.e. if $L' = L \cup \{p\}$, then p is the last element of L'. For a subset $L = \{j_i, \dots, j_k\} \subset J$ we denote by G_L the submatrix $[g_{j_1}, \dots, g_{j_k}]$ of the matrix G, by b_L – the vector $(\beta_{j_1}, \dots, \beta_{j_k})^\top$ and by r_L – the vector $(\rho_{j_1}, \dots, \rho_{j_k})^\top$.

Algorithm 6.

Input:

a) a system of linear inequalities $G^\top x \le b$,

b) an approximation $\overline{x}$ of a solution to this system.

Output:

a new approximation x^+ of a solution or inconsistency detection.

Step 0. (*Initialization*)

0.1. Evaluate the residual vector $r = (\rho_1, \dots, \rho_m)^\top = G^\top \overline{x} - b$,

0.2. Set $L = \{i\}$ for $i \in J$ such that $\rho_i > 0$; If such i does not exist, then print: "$\overline{x}$ is a solution" and go to Step 6.

0.3. Set $G_L = g_i$.
0.4. Set $C_L = \|g_i\|$.

Step 1. Set $K = \emptyset$.

Step 2. (*Stopping criterion*) If $L \cup K = J$ go to Step 5.

Step 3. (*Choice of a trial inequality*) Choose any $p \in J \setminus (L \cup K)$.

Step 4. (*Residual selection*)
4.1. Evaluate w as the unique solution to the system $C_L C_L^\top w = G_L^\top g_p$,
4.2. If $w \le 0$ and $w^\top r_L \le \rho_p$, then
a) set $L := L \cup \{p\}$,
b) set $G_L := [G_L, g_p]$,
c) make an update of the Cholesky factorization $C_L C_L^\top$ of the matrix $G_L^\top G_L$; if the Cholesky procedure breaks down, then print "$\{x \in \mathbb{R}^n : G^\top x < b\} = \emptyset$" and go to Step 6,
d) go to Step 1.
Otherwise set $K := K \cup \{p\}$ and go to Step 2.

Step 5. (*Projection*) Set $x^+ = \overline{x} - G_L(C_L C_L^\top)^{-1} r_L$.

Step 6. Terminate.

Remark 7. (a) Suppose for a moment that $\{x \in \mathbb{R}^n : G^\top x < b\} \neq \emptyset$. The conditions in Step 4.2 of Algorithm 6 correspond to conditions (iii) and (iv) in Theorem 3. Furthermore, observe that the conditions (i) and (ii) in Theorem 3 are also satisfied for a matrix G_L constructed in Algorithm 6. For the initial matrix $G_L = g_i$ these conditions are satisfied since $i \in J$ is such that $\rho_i > 0$ in Step 0.2 (one can easily show that g_i is nonzero in this case); For the matrix G_L constructed in Step 4.2b) these conditions are satisfied by Theorem 3 (see (v) and (vi)). Now we see that if the Cholesky procedure detects the linear dependency of the columns of G_L in Step 4.2c), then we obtain a contradiction, which proves by Theorem 3 that $\{x \in \mathbb{R}^n : G^\top x < b\} = \emptyset$. In other case the matrix G_L has a full column rank and, by Corollary 5, the vector x^+ determined in Step 5 is the projection of $\overline{x}$ onto the subset $\{x : G_L^\top x \le b_L\}$.

(b) If we suppose that $G^\top \overline{x} \ge b$, then Algorithm 6 realizes the obtuse cone selection model. Observe that in this case the assumptions of Theorem 1 are satisfied in each iteration of the Algorithm for $A_1 = G_L$ and for $a = g_p$. It follows from Theorem 1 that the condition $w^\top r_L \le \rho_p$ in Step 4.2 holds automatically (and can be skipped) in this case since

$$w = (C_L C_L^\top)^{-1} G_L^\top g_p = (G_L G_L^\top)^{-1} G_L^\top g_p = G_L^+ g_p = A_1^+ a \le 0$$

and $r = G^\top \overline{x} - b \geq 0$. Furthermore, if we replace the condition $w \leq 0$ in Step 4.2 by the condition $G_L^\top g_p \leq 0$, then the Algorithm realizes the regular obtuse cone selection model (see Remark 2). We can drop Step 4.1 in this case. Furthermore, in this case we can replace Step 4.2d) in Algorithm 6 by 4.2d') go to Step 2 since a regular obtuse cone has the following hereditary property: If the cone generated by the columns of G_L is a regular obtuse cone and $L' \subset L$, then $G_{L'}$ generates also a regular obtuse cone.

3.3. Application to projection methods

In projection methods (3) for convex minimization problems the system of linear inequalities $G^\top x \leq b$ described in the previous Section often has the following form:

$$f_i(x) = g_i^\top (x - x_i) + f(x_i) \leq \alpha_k, \quad i \in J_k, \tag{16}$$

where $J_k \subset \{1, \ldots, k\}$ is a subset of saved linearizations and the level α_k is an approximation of the minimal value f^* of f. The value α_k is given in the form $\alpha_k = (1 - \nu)\overline{\alpha}_k + \nu \underline{\alpha}_k$, where $\overline{\alpha}_k = \min_{i \leq k} f(x_i)$, $\underline{\alpha}_k$ is a known lower bound of f^* and the level parameter $\nu \in (0, 1)$. Suppose we employ the residual selection method for the system (16) in each iteration of the projection method. The selection of a subsystem $G_L^\top x \leq b_L$ is equivalent in this case to the selection of L_k from the subset of saved linearizations J_k. Suppose we know an initial lower bound $\underline{\alpha}_1$ of the minimal objective value f^* and an upper bound δ of $\mathrm{d}(x_1, M)$ for a starting point x_1. If we detect that the inequality converse to (5) holds then, of course, $\alpha_k \leq f^*$, as we have seen in Section 2. In this case, we can take α_k as a new lower bound of f^* (such a level update was first used in [7] and was also applied in [8, 2]). Furthermore, if we apply the presented residual selection model to the projection method (3), we can detect that $\alpha_k \leq f^*$ in an other way. Suppose that $\alpha_k > f^*$. Then, of course, $f_i(x^*) < \alpha_k$, $i \in J_k$ for $x^* \in M$, and an application of any selection model presented in Sections 3.1 and 3.2 leads to a linearly independent system $\{g_i : i \in L_k\}$. Therefore, if we detect that the selected subgradients are linearly dependent, then we are sure that the assumption $\alpha_k > f^*$ is not true. This linear dependence can be detected by the Cholesky procedure in Step 4c) of Algorithm 6. In this case, a lower bound of f^* can be updated by setting $\underline{\alpha}_{k+1} = \alpha_k$. A similar observation was employed in [2, Section 4] and in [9, Section 5]. Numerical experiments

show that among the presented two possibilities the inequality $\alpha_k \leq f^*$ is detected in most cases by the Cholesky procedure.

The convergence of the described method to a solution x^* follows from the results presented in [8, 2] which concern a more general model.

3.4. Order of checked subgradients in selection strategies

The choice of a vector g_p in the presented residual selection model described by Algorithm 6 corresponds to a choice of g_p from the saved subgradients which should be checked. The following strategies of a choice of p in Step 3 of this Algorithm seems to be reasonable:

a) Reverse iteration's order strategy:
   - Choose the largest $p \in J \setminus (L \cup K)$.

b) Largest residuum strategy:
   - Choose $p \in J \setminus (L \cup K)$ which corresponds to a linearization with the largest residuum, i.e.

$$p = \operatorname{argmax}\{g_i^\top \overline{x} - \beta_i : i \in J \setminus (L \cup K)\}$$

   (of course $\rho_p = f_p(x_k) - \alpha_k$).

c) Furthest inequality strategy:
   - Choose $p \in J \setminus (L \cup K)$ which corresponds to the furthest inequality, i.e.

$$p = \operatorname{argmax}\{\|P_{\{x:g_i^\top x \leq \beta_i\}}(\overline{x}) - \overline{x}\| : i \in J \setminus (L \cup K)\}$$

   (of course

$$\begin{aligned} s_p \quad &: \quad = P_{\{x:g_p^\top x \leq \beta_p\}}(\overline{x}) - \overline{x} \\ &= \quad \frac{g_p^\top \overline{x} - \beta_p}{\|g_p\|^2} g_p = \frac{f_p(x_k) - \alpha_k}{\|g_p\|^2} g_p) \end{aligned}$$

d) Largest projection vector strategy:
   - Choose $p \in J \setminus (L \cup K)$ which corresponds to a linearization with the largest projection vector

$$t = -G_L (G_L^\top G_L)^{-1} r_L.$$

In the first three strategies it is enough to order the subset J with respect to a corresponding strategy. This order generates a corresponding hereditary order of $J \setminus (L \cup K)$. The fourth strategy has not such a heredity property. Therefore, in the fourth strategy each realization of Step 3 of Algorithm 6 requires a selection of a corresponding $p \in J \setminus (L \cup K)$. The following Lemma will be useful in order to select the largest projection vector.

Lemma 8. *If the assumptions of Theorem* 3 *are satisfied, then*

$$\|t\|^2 = \|t_1\|^2 + \frac{[\rho - (A_1^+ a)^\top r_1]^2}{\|a^\perp\|^2}, \tag{17}$$

where $t = P_{\{x:A^\top x \le b\}}(\overline{x}) - \overline{x}$, $t_1 = P_{\{x:A_1^\top x \le b_1\}}(\overline{x}) - \overline{x} = -A_1(A_1^\top A_1)^{-1} r_1$ *and* $a^\perp = (I - A_1 A_1^+)a$.

Proof. By Corollary 5 and by the definition of the residuum r given by (10) we have $t = -A(A^\top A)^{-1} r$. Now, by equality (13) we obtain

$$\begin{aligned}
\|t\|^2 &= t^\top t = r^\top (A^\top A)^{-1} r \\
&= \left[r_1^\top, \rho \right] \begin{bmatrix} (A_1^\top A_1)^{-1} r_1 - \| a^\perp\|^{-2} [\rho - (A_1^+ a)^\top r_1] A_1^+ a \\ \| a^\perp\|^{-2} [\rho - (A_1^+ a)^\top r_1] \end{bmatrix} \\
&= r_1^\top (A_1^\top A_1)^{-1} r_1 - \|a^\perp\|^{-2} [\rho - (A_1^+ a)^\top r_1] r_1^\top A_1^+ a \\
&\quad + \|a^\perp\|^{-2} \rho [\rho - (A_1^+ a)^\top r_1] \\
&= \|t_1\|^2 + \frac{[\rho - (A_1^+ a)^\top r_1]^2}{\|a^\perp\|^2}.
\end{aligned}$$

■

Remark 9. The last term on the right side of (17) has in the notation of Algorithm 6 in the following form:

$$\frac{[\rho_p - (G_L^+ g_p)^\top r_L]^2}{g_p^\top (I - G_L G_L^+) g_p}.$$

A simple calculation shows that this expression is equal to

$$\frac{[\rho_p - (G_L^\top g_p)^\top (G_L^\top G_L)^{-1} r_L]^2}{\|g_p\|^2 - (G_L^\top g_p)^\top (G_L^\top G_L)^{-1} (G_L^\top g_p)} \,. \tag{18}$$

Now it follows from Lemma 8 that it is sufficient to select $p \in J \setminus (L \cup K)$ which maximizes the above expression in order to select the largest projection vector.

4. Numerical tests

In this Section, we present the computation results of various variants of the presented projection method with residual selection for convex minimization problems, and we compare these results with other methods: with the method of projection onto an acute cone [2] (where an obtuse cone selection was employed), with the bundle trust region method [14] and with the variable target value subgradient method [7].

4.1. Test problems

In our numerical experiments we have tested the methods presented in the previous Section for the following test problems:

1. Shor's test problem (Shor) [13]

$f(x) = \max_{1 \le i \le 10} b_i \sum_{j=1}^{n} (\xi_j - a_{ij})^2$, $n = 5$, $f^* \cong 22.600162095771$, $\|x_1 - x^*\| \cong 2.2955$, $f(x_1) = 80$.

2. Goffin's test problem (Goffin)

$f(x) = n \max_{1 \le j \le n} \xi_j - \sum_{j=1}^{n} \xi_j$, $n = 50$, $f^* = 0$, $\|x_1 - x^*\| \cong 102.042$, $f(x_1) = 1225$.

3. Hilbert's test problem (L1hil)

$f(x) = \sum_{i=1}^{n} \left| \sum_{j=1}^{n} (\xi_j - 1) / (i + j - 1) \right|$, $n = 10$, $f^* = 0$, $\|x_1 - x^*\| \cong 3.162$, $f(x_1) \cong 13.3754$.

4. Lemaréchal's test problem (Maxquad) [11]

$f(x) = \max_{1\le i\le 5}(x^\intercal A_i x - b_i^\intercal x)$, $n = 10$, $f^* \cong -0.841408334596$, $\|x_1 - x^*\| \cong 3.189$, $f(x_1) = 5337$.

5. Nemirovski's test problem (BadGuy)

$n = 100$, $\|x_1 - x^*\| \le 10240$, $f(x_1) = -1792$.

This test is organized so as to answer the worst possible function/subgradient-values as long as possible. An experiment with this test problem is not reproducible (except if the function and a subgradient are evaluated at the same sequence of points x_k) because the answer at current call depends on the previous calls. We point out that we have obtained in our tests other minimal values than the declared in the comment lines of a subroutine BadGuy.

6. Rosen-Suzuki test problem (Rosen) [5]

$f(x) = \max_{1\le i\le 4}(x^\intercal A_i x - b_i^\intercal x + c_i)$, $n = 4$, $f^* = -44$, $\|x_1 - x^*\| \cong 2.44954$, $f(x_1) = 0$.

7. Lemaréchal's test problem (TR48) [11]

$f(x) = \sum_{i=1}^{n} d_i \max_{1\le j\le n}(\xi_j - a_{ij}) - \sum_{j=1}^{n} s_j \xi_j$, $n = 48$, $f^* = -638565$, $\|x_1 - x^*\| \cong 1978.3889$, $f(x_1) = -464816$.

8. Randomly generated strongly convex problems

$f(x) = \max_{1\le i\le m}(a_i^\intercal x + b_i) + \sum_{j=1}^{n} (\xi_j - c_j)^2$, $n = 5,\ 20,\ 30,\ 50$, $m = 10,\ 20,\ 50,\ 100$.

In these problems, a_i, b_i $(i = 1, \ldots, m)$, are randomly generated in the interval $[-1, 1]$, $c_j, j = 1, \ldots, n$, are randomly generated with entries in the interval $[-2, 2]$.

4.2. Results of numerical tests

Now we present the results of numerical tests for the following methods:

- RS – the method of projection with level control and residual selection with various order of the checked subgradients:
  - (a) reverse iteration's order strategy,
  - (b) largest residuum strategy,
  - (c) furthest inequality strategy,
  - (d) largest projection vector strategy,
- OCS(a) – the method of projection with level control and obtuse cone selection with reverse iteration's order strategy [2],
- ROCS(a) – the method of projection with level control and regular obtuse cone selection with reverse iteration's order strategy [2],
- BT – the bundle trust region method [14],
- VTV – the variable target value subgradient method [7].

In the presented results, we employ various upper bounds δ of $\mathrm{d}(x_1, M) = \inf_{y \in M} \|y - x_1\|$ and we set $D = B(x_1, \delta)$. Furthermore, we employ various initial lower bounds $\underline{\alpha}_1$ of f^*. In all tests the stopping criterion $\overline{\alpha}_k - \underline{\alpha}_k \leq \varepsilon$ was employed with the absolute optimality tolerance ε. In all tables $\#f/g$ denotes the number of objective and subgradient evaluations. The methods were programmed in Fortran 90 (Lahey Fortran 90 v.3.5). All floating point calculations were performed in double precision, allowing the relative accuracy of $2.2 \cdot 10^{-16}$. We set the number of stored subgradients $\#J_k = 100$ for problems 1–6 and 8 (beyond the results presented in Table 10) and $\#J_k = 500$ for Problem 7. All CPU times are given in seconds.

In Table 1, we compare various projection methods with the bundle trust region method for various test problems. We set the relaxation parameter $\lambda = 1$, the level parameter $\nu = 0.5$ and the optimality tolerance $\varepsilon = 10^{-6}$. A good behavior of the RS(a) method with respect to the other methods can be observed. The VTV method is not able to obtain the optimality tolerance better than 10^{-2} in a reasonable number of iterations for almost all test problems. Surprisingly, this method gives an ε-optimal solution for the Nemirovski's test problem (BadGuy) in CPU time which is comparable with CPU time obtained for the RS(a) and OCS(a) methods. Another surprise is that the BT method does not converge for this test problem.

Table 1

↓ problem	method →		RS(a)		OCS(a)		BT		VTV	
	$\underline{\alpha}_1$	δ	$\#f/g$	time	$\#f/g$	time	$\#f/g$	time	$\#f/g$	time
Shor	0	10^2	41	<0.1	54	<0.1	53	0.2	$>10^6$	–
Goffin	-10^2	10^3	66	0.8	77	2.8	53	0.3	$>10^6$	–
L1hil	-10^2	10^3	38	<0.1	43	<0.1	10	<0.1	$>10^6$	–
Maxquad	–10	10^2	150	0.1	339	0.1	439	0.3	$>10^6$	–
BadGuy	-10^4	$2*10^4$	102	0.3	105	0.9	$>10^4$	–	3264	0.6
Rosen	-10^2	10^2	45	<0.1	72	<0.1	2787	0.4	$>10^6$	–
TR48	$-7*10^5$	$5*10^3$	2377	114.0	$>10^4$	–	640	41.4	$>10^6$	–

In Table 2, we present the results of numerical tests for 4 variants of the residual selection method (RS) and for various test problems. We set $\lambda = 1$, $\nu = 0.5$, $\varepsilon = 10^{-6}$. The smallest number of function and subgradient evaluations were obtained for the largest projection vector strategy (variant (d)). Unfortunately, CPU time is essentially longer for this variant with respect to other ones. These not surprising results are due to the necessity of evaluation of (18) for all $p \in I \setminus (L \cup K)$ in this variant (see Remark 9).

Table 2

↓ problem	variant →		RS(a)		RS(b)		RS(c)		RS(d)	
	$\underline{\alpha}_1$	δ	$\#f/g$	time	$\#f/g$	time	$\#f/g$	time	$\#f/g$	time
Shor	0	10^2	41	<0.1	42	<0.1	42	<0.1	39	<0.1
Goffin	-10^2	10^3	66	0.8	66	0.6	66	0.6	66	6.3
L1hil	-10^2	10^3	38	<0.1	44	<0.1	33	<0.1	27	<0.1
Maxquad	–10	10^2	150	0.1	135	0.1	130	0.1	120	0.3
BadGuy	-10^4	$2*10^4$	102	0.3	102	0.6	102	0.3	102	0.6
Rosen	-10^2	10^2	45	<0.1	40	<0.1	40	<0.1	40	<0.1
TR48	$-7*10^5$	$5*10^3$	2377	114.0	4424	697.3	3879	557.4	2005	374.3

In Table 3, we compare the number of objective and subgradient evaluations and CPU time for 4 projection methods. As we have observed in Table 1 the optimality tolerance $\varepsilon = 10^{-6}$ and the parameter δ were too high for the method VTV. In order to present the comparison of CPU time for all 4 projection methods, the upper bound δ is set close to $\|x_1 - x^*\|$ and the optimality tolerance $\varepsilon = 10^{-2}$. Furthermore, we set $\lambda = 1$, $\nu = 0.5$.

The number of objective and subgradient evaluations is essentially smaller for the RS(a) and OCS(a) methods than the for the ROCS(a) and VTV methods, although the CPU time for the two last methods are not so bad in comparison with $\#f/g$. This observation is also not surprising. The cost of 1 iteration for the ROCS method is essentially smaller in comparison with the RS and OCS methods because of the heredity property of the ROCS method (see Remark 7). Furthermore, the cost of 1 iteration for the VTV method is extremely small.

Table 3

↓ problem	method →		RS(a)		OCS(a)		ROCS(a)		VTV	
	$\underline{\alpha}_1$	δ	$\#f/g$	time	$\#f/g$	time	$\#f/g$	time	$\#f/g$	time
Shor	0	3	20	<0.1	25	<0.1	19488	3.2	z57778	1.0
Goffin	-10^2	105	58	0.4	64	0.7	64	0.2	$>10^6$	–
L1hil	-10^2	4	12	<0.1	10	<0.1	1695	0.3	168077	7.4
Maxquad	–10	4	59	<0.1	138	<0.1	14901	3.3	36140	1.5
BadGuy	-10^4	12000	102	0.2	105	0.5	168	1.3	2088	0.4
Rosen	-10^2	4	20	<0.1	37	<0.1	12672	2.1	42388	0.8
TR48	$-7*10^5$	2000	1713	109.8	$>10^4$	–	$>10^4$	–	$>10^6$	–

In Table 4, we compare the number of objective and subgradient evaluations and CPU time for 3 projection methods, for various test problems and for lower bounds $\underline{\alpha}_1$ equal to f^*. Furthermore, we set $\lambda = 1$, $\nu = 1 - 10^{-6}$, $\varepsilon = 10^{-6}$. Again the best results were obtained for the method RS(a).

Table 4

↓ problem	method →		RS(a)		OCS(a)		ROCS(a)	
	$\underline{\alpha}_1$	δ	$\#f/g$	time	$\#f/g$	time	$\#f/g$	time
Shor	22.60016210	10^2	39	<0.1	39	<0.1	$>10^4$	–
Goffin	0	10^3	51	0.2	51	0.1	51	0.1
L1hil	0	10^3	11	<0.1	12	<0.1	$>10^4$	–
Maxquad	-0.84140833	10^2	42	<0.1	43	<0.1	$>10^4$	–
BadGuy	-2048	$2*10^4$	155	0.4	159	3.5	159	0.9
Rosen	-44	10^2	29	<0.1	29	0.1	$>10^4$	–
TR48	-638565	$5*10^3$	643	26.6	3546	296.0	$>10^4$	–

In Table 5, we present the results of numerical tests for the Shor test problem and for various optimality tolerances.

We set $\underline{\alpha}_1 = 0$, $\delta = 100$, $\lambda = 1$, $\nu = 0.5$. The results for the first three methods are comparable if $\varepsilon \geq 10^{-8}$. A linear convergence is observed for the RS(a) and OCS(a) methods. Unexpectedly, the BT method does not produce even if employed for a long time an essentially better solution after obtaining the accuracy 10^{-8}. Perhaps, the restart of the BT-algorithm can help to obtain faster an ε-optimal solution for $\varepsilon \leq 10^{-8}$ but we have not tested such a modification. As observed before, the VTV method is not able to give an ε-optimal solution for $\varepsilon < 10^{-2}$. A similar behavior of the RS, OCS, BT and VTV methods was also observed for other test problems.

Table 5

method $\rightarrow$	RS(a)	OCS(a)	BT	VTV
ϵ	$\#f/g$	$\#f/g$	$\#f/g$	$\#f/g$
10^{-2}	22	30	28	598124
10^{-4}	31	43	39	$>10^6$
10^{-6}	41	54	53	–
10^{-8}	47	71	70	–
10^{-10}	57	79	3469	–
10^{-12}	70	84	3469	–

In Table 6, we present the influence of the relaxation parameter λ on the convergence of 2 projection methods for the Shor test problem. We set $\underline{\alpha}_1 = 0, \delta = 100, \nu = 0.5$, $\varepsilon = 10^{-6}$. As we see, this influence is not essential for the RS(a) method. The OCS(a) method does not converge for the Shor test problem if $\lambda > 1$. A similar influence is observed in other test problems.

Table 6

method $\rightarrow$	RS(a)	OCS(a)
λ	$\#f/g$	$\#f/g$
0.8	39	55
0.9	41	58
1.0	41	54
1.1	45	$>10^4$
1.2	45	$>10^4$
1.5	44	$>10^4$

In Table 7, we present the influence of the level parameter ν on the convergence of 2 projection methods for the Shor test problem. We set $\underline{\alpha}_1 = 0$, $\delta = 100$, $\lambda = 1$, $\varepsilon = 10^{-6}$. The results show that the influence is not big for both methods.

Table 7

method $\rightarrow$	RS(a)	OCS(a)
ν	$\#f/g$	$\#f/g$
0.3	41	53
0.4	31	50
0.5	41	54
0.6	41	55
0.7	44	63

In Table 8, we present the influence of the upper bound δ of $\mathrm{d}(x_1, M)$ for various variants of the RS method and for various test problems. The second value of δ is chosen close to $\|x_1 - x^*\|$. Furthermore, we set $\lambda = 1$, $\nu = 0.5$, $\varepsilon = 10^{-6}$. Although better results were obtained for the second value of δ, the influence of this parameter on the convergence is not big. The cause of such a behavior is that in almost all cases the inequality $\alpha_k < f^*$ is detected by the Cholesky procedure in Step 4.2c) of Algorithm 6. The inequality converse to 5 holds very rarely evens if δ is chosen close to $\|x_1 - x^*\|$.

Table 8

$\downarrow$ problem	method $\rightarrow$		RS(a)	RS(b)	RS(c)	RS(d)
	δ	$\underline{\alpha}_1$	$\#f/g$	$\#f/g$	$\#f/g$	$\#f/g$
Shor	10^2	0	41	42	42	39
	3		39	40	40	37
Goffin	10^3	-10^2	66	66	66	66
	105		66	66	66	66
L1hil	10^3	-10^2	38	44	33	27
	4		38	29	39	30
Maxquad	10^2	-10	150	135	130	120
	4		137	127	121	119
BadGuy	10^5	-10^4	102	102	102	102
	12000		102	102	102	102
Rosen	100	-10^2	45	40	40	40
	4		41	38	38	36

In Table 9, we present the influence of the lower bound $\underline{\alpha}_1$ of the minimal objective value for various projection methods and for various test problems. We set $\nu = 0.5$ for the first initial lower bound $\underline{\alpha}_1$ of f^* and $\nu = 1 - 10^{-6}$ if $\underline{\alpha}_1 = f^*$. Furthermore, we set $\lambda = 1, \varepsilon = 10^{-6}$. We see that in almost all cases the results are not essentially better for a better initial lower bound $\underline{\alpha}_1$ of f^*.

Table 9

↓ problem	method →		RS(a)	RS(b)	RS(c)	RS(d)
	$\underline{\alpha}_1$	δ	$\#f/g$	$\#f/g$	$\#f/g$	$\#f/g$
Shor	-10^6	10^2	49	43	43	48
	22.60016210		39	39	39	39
Goffin	-10^6	10^3	64	64	64	64
	0		51	51	51	51
L1hil	-10^6	10^3	42	38	37	30
	0		11	11	11	9
Maxquad	-10^6	10^2	181	183	169	154
	−0.84140833		42	42	42	42
BadGuy	-10^6	$2*10^4$	102	102	102	102
	−2048		155	132	133	130
Rosen	-10^6	10^2	53	52	52	50
	−44		29	29	29	29
TR48	−700000	$5*10^3$	2377	4424	3879	2005
	−638565		643	1571	1821	638

In Table 10, we present the influence of the number of stored subgradients on the convergence of the RS(d) method for the Shor problem. Note that the case $\#J_k = 1$ corresponds to the variable target value subgradient method [7]. We set $\delta = 3, \lambda = 1$. The influence is not big if $\#J_k \geq 5$. The number of objective and subgradient evaluations increases rapidly if $\#J_k < 5$. Note that in a lot of iterations the RS method tries to construct a full-dimensional system G_L in Step 4.2b) of Algorithm 6 for the Shor test problem. A construction of such a system is impossible if the number of stored subgradients is less than the dimension of the problem (= 5 in the case under consideration).

Table 10

			$\#f/g$				
ν	$\underline{\alpha}_1$	ε	$\#J_k$= 50	$\#J_k$= 5	$\#J_k$= 4	$\#J_k$= 3	$\#J_k$= 1
0.5	0	10^{-2}	20	18	48	9615	57788
		10^{-4}	29	40	213	$>10^6$	$>10^6$
		10^{-6}	37	57	424	–	–
1.0	f^*	10^{-2}	18	18	18	316	1713
		10^{-4}	29	29	30	30320	170493
		10^{-6}	39	39	37	$>10^6$	$>10^6$

4.3. Numerical results for strongly convex problems

Now we present the numerical results for strongly convex problems of the form $f(x) = \max_{1\le i\le m}(a_i^\intercal x + b_i) + s\sum_{j=1}^n (\xi_j - c_j)^2$, where a_i, b_i $(i = 1,\dots,m)$, are randomly generated in the interval $[-1,1]$, $c_j, j = 1,\dots,n$, are randomly generated with entries in the interval $[-2,2]$ and the strong convexity constant $s = 1$ (Problem 8). Such problems were tested in [7] for the VTV method. It follows from the strong convexity of f that for such problems an upper bound δ of $\operatorname{diam} D$ and a lower bound $\underline{\alpha}_k$ of f^* can be additionally updated in each iteration. More precisely, we can take $\delta_k = \min\{\sqrt{\frac{f(x_k)-\underline{\alpha}_k}{s}}, \frac{\|g_k\|}{s}\}$ as an upper bound of $\operatorname{diam} D$ and we can use an additional information:

$$f^* \ge f(x_k) - \frac{\|g_k\|^2}{2s}$$

in order to update a lower bound $\underline{\alpha}_k$ of f^* (see [7] for details). We call such a modification of the projection method the SC (strongly convex) variant. Since Problem 1 is also strongly convex with the strong convexity constant $s = 1$, we present additionally the numerical results for this problem.

The numerical results for Problem 8 are presented in Table 11. Two methods are compared: the RS(a) method (the basic variant and the strongly convex variant) and the VTV method (the strongly convex variant). We set $\lambda = 1$, $\nu = 0.5$. We have tested only the strongly convex variant of the VTV method since the basic variant of the method is not able to attain the optimality tolerance better than 10^{-2} in a reasonable number of iterations. Contrary to the VTV method, we observe only small differences between the

results of both variants (basic and strongly convex) of the RS(a) method. Note that in almost all iterations of both variants of the RS(a) method the inequality $\alpha_k < f^*$ is detected by the Cholesky procedure in Step 4.2c) of Algorithm 6. For the VTV method the inequality $\alpha_k < f^*$ is detected only by the detection that the inequality converse to 5 holds. This detection requires many iterations since for the VTV method the vectors t_k are extremely short.

Table 11

		$\#f/g$		$\#f/g$
method $\rightarrow$		RS(a)		VTV
$m \times n$	ε	basic variant	SC variant	SC variant
10×5	10^{-2}	12	9	38
	10^{-4}	16	14	3068
	10^{-6}	20	18	>3*10^4
20×20	10^{-2}	17	12	190
	10^{-4}	23	16	12472
	10^{-6}	28	21	>3*10^4
50×30	10^{-2}	14	10	70
	10^{-4}	18	14	6149
	10^{-6}	23	18	>3*10^4
100×50	10^{-2}	18	19	268
	10^{-4}	23	24	12328
	10^{-6}	27	29	>3*10^4

Numerical results for the Shor test problem (Problem 1) are presented in Table 12. Three methods are compared: the RS method, the OCS method (in each case the basic variant and the strongly convex variant), and the VTV method (the strongly convex variant). We set $\lambda = 1$, $\nu = 0.5$. The basic variant of the VTV method requires about $6 * 10^5$ iterations to obtain the accuracy 10^{-2} (see Table 5) and therefore was not considered in the current test. Similarly as in the results presented in Table 11, only small differences between the results of both variants (basic and strongly convex) of the RS(a) and OCS(a) methods were observed.

Table 12

	$\#f/g$		$\#f/g$		$\#f/g$
method $\rightarrow$	RS(a)		OCS(a)		VTV
ε	basic variant	SC variant	basic variant	SC variant	SC variant
10^{-2}	22	23	30	27	6204
10^{-4}	31	32	43	40	34692
10^{-6}	41	37	54	51	$>5*10^4$

4.4. Conclusions

The numerical experiments show that all variants of the projection method with residual selection converge linearly for the tested problems. For these problems the number of objective and subgradient evaluations for the presented OCS and RS methods and for the bundle trust region method (BT) [14] are comparable if $\varepsilon \geq 10^{-6}$. However, the results show the superiority of the presented OCS and RS methods in comparison with BT method for higher optimality tolerances. A disadvantage of the OCS and RS methods is the fact that an upper bound δ of $\mathrm{d}(x_1, M)$ and an initial lower bound $\underline{\alpha}_1$ of the minimal objective value f^* must be known. Nevertheless, the results presented in Tables 8 and 9 show that the values of these bounds do not have an essential influence on the speed of convergence. The results presented in Table 10 show an essential influence of the number of stored linearizations on the speed of convergence. Note that for the tested Shor problem any version of the presented projection method very often constructs a 5-element subsystem of the stored linearizations if the number of the stored linearization equals at least 5. Therefore, in other cases the speed of convergence goes down. For all the tested functions the projection method with residual selection behaves better than the with the obtuse cone selection. The numerical tests show that the best results were obtained for the projection method with residual selection and with the largest projection vector strategy (RS(d)) – see results presented in Tables 2, 8 and 9. Note, however, that the RS(d) method is the most time consuming per iteration because of the necessity of evaluation of (18) for all $p \in I \setminus (L \cup K)$ (see Remark 9). The results presented in Tables 11 and 12 show the superiority of the projection method with residual selection in comparison with the variable target value

method for strongly convex functions (see [7], where also numerical results for similar randomly generated function were presented). Note that the last method is a special case of the projection method with residual selection, where the number of stored subgradients is equal to 1.

It is interesting that all presented projection methods (RS, OCS, ROCS and VTV) have the same theoretical efficiency – they give an ε-optimal solution in at most $(\delta L/\varepsilon)^2$ objective and subgradient evaluations, where δ is an upper bound of $\mathrm{d}(x_1, M)$ and L is a Lipschitz constant of f (see [8, 2]). From this point of view, such a huge difference in the numerical results between the RS, OCS, ROCS methods and the VTV method is surprising.

Acknowledgment

We wish to thank to Michal Kočvara for sending us a Fortran subroutine which realizes the BT-method and to Krzysztof C. Kiwiel for sending us Fortran subroutines which evaluate objective/subgradient values for test problems 1–7.

References

[1] A. Cegielski, *Projection onto an acute cone and convex feasibility problems*, J. Henry and J.-P. Yvon (eds.), Lecture Notes in Control and Information Science **197**, Springer-Verlag, London (1994), 187–194.

[2] A. Cegielski, *A method of projection onto an acute cone with level control in convex minimization*, Mathematical Programming **85** (1999), 469–490.

[3] A. Cegielski, *Obtuse cones and Gram matrices with non-negative inverse*, Linear Algebra and its Applications **335** (2001), 167–181.

[4] A. Cegielski and R. Dylewski, *Residual selection in a projection method for convex minimization problems*, (submitted).

[5] J. Charalambous and A.R. Conn, *An efficient method to solve the minimax problem directly*, SIAM J. Num. Anal. **15** (1978), 162–187.

[6] R. Dylewski, *Numerical behavior of the method of projection onto an acute cone with level control in convex minimization*, Discuss. Math. Differential Inclusions, Control and Optimization **20** (2000), 147–158.

[7] S. Kim, H. Ahn and S.-C. Cho, *Variable target value subgradient method*, Mathematical Programming **49** (1991), 359–369.

[8] K.C. Kiwiel, *The efficiency of subgradient projection methods for convex optimization, part* I: *General level methods*, SIAM J. Control and Optimization **34** (1996), 660-676.

[9] K.C. Kiwiel, *The efficiency of subgradient projection methods for convex optimization, part* II: *Implementations and extensions*, SIAM J. Control and Optimization **34** (1996), 677–697.

[10] K.C. Kiwiel, *Monotone Gram matrices and deepest surrogate inequalities in accelerated relaxation methods for convex feasibility problems*, Linear Algebra and Applications **252** (1997), 27–33.

[11] C. Lemaréchal and R. Mifflin, *A set of nonsmooth optimization test problems*, Nonsmooth Optimization, C. Lemarechal, R. Mifflin, (eds.), Pergamon Press, Oxford 1978, 151–165.

[12] C. Lemaréchal, A.S. Nemirovskii and Yu.E. Nesterov, *New variants of bundle methods*, Math. Progr. **69** (1995), 111–147.

[13] N.Z. Shor, Minimization Methods for Non-differentiable Functions, Springer-Verlag, Berlin 1985.

[14] H. Schramm and J. Zowe, *A version of the bundle idea for minimizing of a nonsmooth function: conceptual idea, convergence analysis, numerical results*, SIAM J. Control and Optimization **2** (1992), 121–152.

[15] M.J. Todd, *Some remarks on the relaxation method for linear inequalities*,. Technical Report 419 (1979), Cornell University.

Received 20 March 2002