ISSN 1990-4789, Journal of Applied and Industrial Mathematics, 2019, Vol. 13, No. 3, pp. 557–574. © Pleiades Publishing, Ltd., 2019.
Russian Text © The Author(s), 2019, published in Diskretnyi Analiz i Issledovanie Operatsii, 2019, Vol. 26, No. 3, pp. 60–86.

Adaptive Mirror Descent Algorithms for Convex and Strongly Convex Optimization Problems with Functional Constraints

F. S. Stonyakin[1,2*], **M. Alkousa**[2**], **A. N. Stepanov**[1***], **and A. A. Titov**[2****]

[1] *Vernadsky Crimean Federal University, pr. Akad. Vernadskogo 4, Simferopol, 295007 Russia*

[2] *Moscow Institute of Physics and Technologies, Institutskii per. 9, Dolgoprudnyi, 141701 Russia*

Received October 17, 2018; in final form, February 24, 2019; accepted February 27, 2019

Abstract—Under consideration are some adaptive mirror descent algorithms for the problems of minimization of a convex objective functional with several convex Lipschitz (generally, nonsmooth) functional constraints. It is demonstrated that the methods are applicable to the objective functionals of various levels of smoothness: The Lipschitz condition holds either for the objective functional itself or for its gradient or Hessian (while the functional itself can fail to satisfy the Lipschitz condition). The main idea is the adaptive adjustment of the method with respect to the Lipschitz constant of the objective functional (its gradient or Hessian), as well as the Lipschitz constant of the constraint. The two types of methods are considered: adaptive (not requiring the knowledge of the Lipschitz constants neither for the objective functional nor for constraints) and partially adaptive (requiring the knowledge of the Lipschitz constant for constraints). Using the restart technique, some methods are proposed for strongly convex minimization problems. Some estimates of the rate of convergence are obtained for all algorithms under consideration in dependence on the level of smoothness of the objective functional. Numerical experiments are presented to illustrate the advantages of the proposed methods for some examples.

DOI: 10.1134/S1990478919030165

Keywords: *adaptive mirror descent method, Lipschitz condition, Lipschitz gradient, Lipschitz Hessian, strongly convex function, the technique of restarts*

INTRODUCTION

The problems of constrained minimization of convex smooth and nonsmooth functionals arise in many areas of modern large-scale optimization and its applications [1, 2]. There are numerous methods for these problems among which we mention the bundle-level method [3] and the penalty function method [4–6]. The mirror descent method (MDM) [7–9] traces its origin to the usual gradient descent and can be well considered as a rather simple method for the problems of nonsmooth convex optimization. The present article is dedicated to some adaptive mirror descent methods for the convex programming problems with Lipschitz constraints.

Note that the constraint functionals, in general, may be nonsmooth (nondifferentiable), and so it is natural to consider subgradient methods. The methods using the subgradients of nonsmooth convex functions have been actively developed for several decades. These studies stem from the well-known pioneering works one of which is devoted to the gradient method for unconditional problems with the Euclidean distance [10], and the other, to its generalization for the problems with constraints [11].

In [11] the idea is proposed of switching steps between the direction of the subgradient of the objective functional and the direction of a subgradient of a constraint. Generalization of the gradient descent

[*] E-mail: fedyor@mail.ru

[**] E-mail: mohammad.alkousa@phystech.edu

[***] E-mail: stepanov.student@gmail.com

[****] E-mail: a.a.titov@phystech.edu

method to the problems with non-Euclidean distance is called the *method of mirror descent*. This method is proposed in [8, 9] for the problems without constraints (see also [7]). The mirror descent for the problems with functional constraints is proposed in [9] (see also [12]). In this case, as a rule, to find the step size and stopping criterion for the mirror descent, it is necessary to know the value of the Lipschitz constant of the objective functional, and also of the functional constraints. There are also methods with adaptive meshsize selection, reviewed in [13] for the problems without constraints and in [12], for the problems with constraints. Recently in [14] some mirror descent algorithms, optimal from the point of view of lower oracle estimates, are proposed for convex programming problems with Lipschitz constraint functionals; these algorithms involve adaptive step selection and adaptive stopping criteria. Modifications of these methods for the problems with several convex functional constraints are analyzed in [15, 16].

In this article, some mirror descent algorithms are presented for the problems of minimization of a convex functional f on some convex closed set with a constraint generated by a convex, Lipschitz and nonsmooth functional $g(x) \leq 0$. It is important that the estimates are obtained for the convergence rate of the methods for objective functionals of various smoothness levels. In particular, the objective functional f may not satisfy the Lipschitz property, but have a Lipschitz-continuous gradient. For example, quadratic functionals do not satisfy the usual Lipschitz property (or the Lipschitz constant is rather large) but they have a Lipschitz-continuous gradient. We can also consider nonsmooth convex functions, equal to the maximum of a finite set of differentiable functionals with a Lipschitz-continuous gradient. For example, let A_i ($i \in \overline{1,m}$) be positively semidefinite matrices ($x^\top A_i x \geq 0$ for every $x \in X$, where X is the domain of the problem); and let the objective functional be

$$f(x) = \max_{i=\overline{1,m}} f_i(x), \tag{1}$$

where

$$f_i(x) = \frac{1}{2}\langle A_i x, x\rangle - \langle b_i, x\rangle + c_i, \qquad i = \overline{1,m}, \tag{2}$$

for some fixed $b_i \in \mathbb{R}^n$ and $c_i \in \mathbb{R}$, for all $i = \overline{1,m}$. Note that functionals of the form (1)–(2) arise in the problems of designing mechanical structures, Truss Topology Design, with weighted bars [17]. For the problems of minimization of functionals of this type in the presence of convex Lipschitz constraints in [14, 15, 18], on the basis of the methodology of Nesterov's works [3, 17], some new adaptive mirror descent algorithms are proposed and their optimality is justified. Some of these results are published in [18]. The present paper is devoted to the exposition of the main results of the report [18], as well as the development of the results of [14, 15, 18] in the following directions:

Firstly, from the point of view of lower oracle estimates, the optimality of the methods of [14, 15, 18] is proved for the problems with convex Lipschitz objective functional as well as for the problems with a Lipschitz Hessian in presence of convex Lipschitz constraints.

Secondly, on the basis of the technique of the restart of methods from [14, 18] (for convex problems), some new mirror descent algorithms are proposed for minimization problems for a μ-strongly convex functional f with nonpositive, μ-strongly convex, and Lipschitz nonsmooth constraint functional g. Note that the technique of restart of a method for convex problems to accelerate convergence for strongly convex problems stems from the 1980s (see [9, 19]). The technique of this type of was used in [20] to substantiate a higher rate of convergence of the method of mirror descent for a strongly convex objective functional in the problems without constraints.

Thirdly, we describe a series of numerical experiments illustrating the advantages of the proposed methods over their analogs. In particular, it is shown that, for the Fermat–Torricelli–Steiner problem (the objective functional satisfies the Lipschitz condition with constant 1) with quadratic constraints, the proposed method can work much faster than a similar adaptive method which is also optimal in terms of lower oracle estimates on a class of problems with Lipschitz objective functional [14, Section 3.1]. The calculations are presented, which illustrate some of the advantages of the proposed methods for strongly convex problems.

The article consists of an introduction and four main sections. In Section 1, we give some auxiliary information, the basic concepts for the mirror descent method, and also describe the adaptive Algorithm 1 of mirror descent from [14, Section 3.3] and partially adaptive Algorithm 2 [18]. In Section 2, we consider new statements about the estimates of the rate of convergence of these methods and justify their optimality on the class of problems under consideration under various assumptions on the level of smoothness of the objective functional. Section 3 is dedicated to methods for minimizing strongly convex functions with restarts of Algorithm 1 (Algorithm 3) and Algorithm 2 (Algorithm 4), as well as the corresponding theoretical estimates for the convergence rate. In Section 4, we give and discuss the results of numerical experiments illustrating certain advantages of the proposed methods.

1. ADAPTIVE AND PARTIALLY ADAPTIVE ALGORITHM OF THE MIRROR DESCENT FOR THE PROBLEMS WITH CONVEX FUNCTIONALS

Let us start with the formulation of the optimization problems under consideration and present the necessary concepts and results. Let $(E, \|\cdot\|)$ be a finite-dimensional normed vector space, and let E^* be a dual space of E with the standard norm

$$\|y\|_* = \max_x\{\langle y, x\rangle,\ \|x\| \le 1\},$$

where $\langle y, x\rangle$ is the value of the continuous linear functional y at $x \in E$.

From now on, we will assume that $X \subset E$ is a closed convex set. Consider two convex subdifferentiable functionals $f, g\colon X \to \mathbb{R}$. Suppose that g satisfies the Lipschitz condition with respect to the norm $\|\cdot\|$; i.e., there is $M_g > 0$ such that

$$|g(x) - g(y)| \le M_g\|x - y\| \tag{3}$$

for all $x, y \in X$. It means that at every point $x \in X$ there is a subgradient $\nabla g(x)$, and $\|\nabla g(x)\|_* \le M_g$. Recall that for a differentiable functional g the subgradient $\nabla g(x)$ coincides with the usual gradient.

In the present work, the following type of optimization problem will be considered:

$$f(x) \to \min, \qquad x \in X, \tag{4}$$

$$g(x) \le 0, \tag{5}$$

if f and g satisfy the above conditions. We also make an assumption about the solvability of problem (4)–(5). Let x_* denote the exact solution of (4)–(5). Our goal is to propose a method that allows us to find some ε-solution $\widehat{x} \in X$ of the problem by finitely many steps:

$$f(\widehat{x}) - f(x_*) \le \varepsilon \qquad \text{for} \quad g(\widehat{x}) \le \varepsilon.$$

Everywhere below we assume that the initial approximation x^0 for all methods is chosen so that

$$d(x^0) = \min_{x\in X} d(x).$$

Note that some results of the present work (Section 3) are devoted to the formulation of the problem for μ-strongly convex subdifferentiable functionals $f, g\colon X \to \mathbb{R}$; i.e.

$$f(y) \ge f(x) + \langle\nabla f(x), y - x\rangle + \frac{\mu}{2}\|y - x\|^2 \qquad \forall\, x, y \in X, \tag{6}$$

and (6) holds for g (with the same strong convexity parameter μ).

For further reasoning, we need some auxiliary concepts (for example, see [13]). Introduce the so-called *prox function* $d : X \to \mathbb{R}$ possessing the property of continuous differentiability and 1-strong convexity with respect to $\|\cdot\|$. Let a constant $\Theta_0 > 0$ be such that $d(x_*) \le \Theta_0^2$. Note that if there is a set of solutions X_* then we assume that

$$\min_{x_*\in X_*} d(x_*) \le \Theta_0^2.$$

Given $x, y \in X$, we consider the corresponding Bregman divergence:

$$V(x, y) = d(y) - d(x) - \langle\nabla d(x), y - x\rangle.$$

Depending on the statement of a specific problem, various approaches are possible for defining the prox structure of the problem and the corresponding Bregman divergence: Euclidean, entropy, and many others (for example, see [13]). We define the projection operator in the standard fashion:

$$\mathrm{Mirr}_x(p) = \arg\min_{u\in X}\big\{\langle p, u\rangle + V(x, u)\big\} \qquad \text{for every}\ \ x \in X\ \ \text{and}\ \ p \in E^*.$$

We assume also that $\mathrm{Mirr}_x(p)$ is easily computable.

Recall the well-known statement (for example, see [13]):

Lemma 1. *Let $f\colon X \to \mathbb{R}$ be a subdifferentiable convex functional on the convex set X and let $z = \mathrm{Mirr}_y(h\nabla f(y))$ for some $y \in X$. Then for all $x \in X$ and $h > 0$ the following holds:*

$$h\langle \nabla f(y), y - x\rangle \leq \frac{h^2}{2}\,\|\nabla f(y)\|_*^2 + V(y, x) - V(z, x). \tag{7}$$

Let us proceed to description of the methods under consideration [14, 18] for problem (4)–(5). Note that it is necessary to find a suitable point $\widehat{x}$ for which $f(\widehat{x}) - f(x_*) \leq \varepsilon$ (or $C \cdot \varepsilon$ for some constant $C > 0$) under the condition $g(x) \leq \varepsilon$.

As can be seen from the listings of the algorithms given in this section, the needed point is selected among the points x^k for which $g(x^k) \leq \varepsilon$. Therefore, we will call step k *productive* if $g(x^k) \leq \varepsilon$. If the reverse inequality $g(x^k) > \varepsilon$ holds then step k will be called *unproductive*. Recall the following algorithm of adaptive mirror descent [14, Section 3.3] for problem (4)–(5):

Algorithm 1 (adaptive mirror descent)

```
Input: accuracy $\varepsilon > 0$; initial point $x^0$; $\Theta_0$; $X$; $d(\cdot)$.
Output: $\bar{x}^N := \arg\min\limits_{x^k,\, k\in I} f(x^k)$.
$I := \varnothing$
$N \leftarrow 0$
repeat
  if $g(x^N) \leq \varepsilon$ then
    $h_N \leftarrow \dfrac{\varepsilon}{\|\nabla f(x^N)\|_*}$
    $x^{N+1} \leftarrow \mathrm{Mirr}_{x^N}(h_N \nabla f(x^N))$        ▷ “productive steps”
    $N \to I$
  else
    $(g(x^N) > \varepsilon) \to$
      $h_N \leftarrow \dfrac{\varepsilon}{\|\nabla g(x^N)\|_*^2}$
      $x^{N+1} \leftarrow \mathrm{Mirr}_{x^N}(h_N \nabla g(x^N))$        ▷ “unproductive steps”
    endif
    $N \leftarrow N + 1$
until $\Theta_0^2 \leq \dfrac{\varepsilon^2}{2}\Big(|I| + \sum\limits_{k\notin I} \dfrac{1}{\|\nabla g(x^k)\|_*^2}\Big)$
```

By analogy with [3], we introduce for the objective functional f at $y \in X$ the following auxiliary quantity:

$$v_f(x, y) = \begin{cases} \Big\langle \frac{\nabla f(x)}{\|\nabla f(x)\|_*}, x - y \Big\rangle, & \nabla f(x) \neq 0, \\ 0, & \nabla f(x) = 0, \end{cases} \qquad x \in X. \tag{8}$$

We allow that, during the method application, an arbitrary subgradient $\nabla f(x)$ can be used.

To estimate the convergence rate of Algorithm 1, the following was proved in [14]:

Theorem 1. *Suppose that inequality* (3) *holds and the constant $\Theta_0 > 0$ is given such that $d(x_*) \leq \Theta_0^2$. If $\varepsilon > 0$ is given then Algorithm* 1 *makes at most*

$$N = \left\lceil \frac{2\max\{1, M_g^2\}\Theta_0^2}{\varepsilon^2} \right\rceil \tag{9}$$

iterations, after that it stops and

$$\min_{k\in I} v_f(x^k, x_*) < \varepsilon. \tag{10}$$

A partially adaptive algorithm can be proposed for Problem (4)–(5) [18]. As distinct from Algorithm 1, the adaptive step is selected only for productive iterations, while the stopping criterion is nonadaptive:

Algorithm 2 (partially adaptive version of Algorithm 1)

```
Input: accuracy $\varepsilon > 0$; initial point $x^0$; $\Theta_0$; $X$; $d(\cdot)$.
Output: $\bar{x}^N := \arg\min_{x^k, k\in I} f(x^k)$.
$x^0 = \arg\min_{x\in X} d(x)$
$I := \varnothing$
$N \leftarrow 0$
repeat
    if $g(x^N) \le \varepsilon$ then
        $h_N \leftarrow \frac{\varepsilon}{M_g \cdot \|\nabla f(x^N)\|_*}$
        $x^{N+1} \leftarrow \mathrm{Mirr}_{x^N}(h_N \nabla f(x^N))$        ▷ “productive steps”
        $N \to I$
    else
        $(g(x^N) > \varepsilon) \to$
        $h_N \leftarrow \varepsilon/M_g^2$
        $x^{N+1} \leftarrow \mathrm{Mirr}_{x^N}(h_N \nabla g(x^N))$        ▷ “unproductive steps”
    endif
    $N \leftarrow N + 1$
until $N \ge \lceil 2M_g^2\Theta_0^2/\varepsilon^2 \rceil$
```

The following analog of Theorem 1 is true (also see [18]):

Theorem 2. *Let $\varepsilon > 0$ be given and let Algorithm 2 make*

$$N = \left\lceil \frac{2M_g^2\Theta_0^2}{\varepsilon^2} \right\rceil \tag{11}$$

iterations. Then

$$\min_{k\in I} v_f(x^k, x_*) < \frac{\varepsilon}{M_g}. \tag{12}$$

Remark 1. Let us consider the situation when the partially adaptive version of the algorithm can be more advantageous than the adaptive algorithm. For example, let there be no possibility to precisely determine the (sub)gradient and thus the norm $\|\nabla g(x^k)\|_*$ for the constraint for one or more unproductive steps, and let only some approximation of this norm be available as $\|\nabla g(x^k)\|_* = \alpha_k \pm \delta_k$, where δ_k is the approximation accuracy. By Lemma 1, at each unproductive step x^k, we have

$$h_k(g(x^k) - g(x_*)) \le \frac{h_k^2}{2}\|\nabla g(x^k)\|_*^2 + V(x^k, x_*) - V(x^{k+1}, x_*). \tag{13}$$

If $\alpha_k = 0$ or $\alpha_k \to 0$ then we cannot use (13) for

$$h_k = \frac{\varepsilon}{\|\nabla g(x^k)\|_*^2}$$

because this may lead to a large error. In this case, the nonadaptive selection of the step

$$h_k = \frac{\varepsilon}{M_g^2}$$

in Algorithm 2 is more suitable for solving problem (4)–(5).

2. ESTIMATES FOR THE CONVERGENCE RATE OF THE METHODS AND THEIR OPTIMALITY

Consider some specific estimates for the convergence rate of the methods under consideration, which justify their optimality from the viewpoint of the theory of oracle estimates, tracing back to the famous monograph by Nemirovski and Yudin [9]. The constraint functionals are assumed to be Lipschitz and, in general, nonsmooth. Therefore, to establish the optimality of the method in terms of lower oracle estimates, we need to show [13] that, to achieve the required accuracy ε of the solution of problem (4)–(5), it suffices to carry out $O(\varepsilon^{-2})$ iterations of the method, and thereafter the calculation of a (sub)gradient of the objective functional or constraint is implemented. In this work, various classes of objective functionals are considered. As before, we denote by x_* the solution to problem (4)–(5). The following auxiliary statement will be used (see, for example, [3, 17]):

Lemma 2. *We introduce the function*

$$\omega(\tau) = \max_{x\in X}\{f(x) - f(x_*) \mid \|x - x_*\| \le \tau\}, \tag{14}$$

where τ is a positive number. Then for every $y \in X$ we have

$$f(y) - f(x_*) \le \omega(v_f(y, x_*)). \tag{15}$$

By Lemma 2, it is shown in [18], how, using Theorem 1, we can estimate the convergence rate of Algorithm 2 if the objective functional f is differentiable and its gradient satisfies the Lipschitz condition:

$$\|\nabla f(x) - \nabla f(y)\|_* \le L\|x - y\| \qquad \forall\, x, y \in X. \tag{16}$$

Then for arbitrary $x \in X$ the following is true [3]:

$$f(x) \le f(x_*) + \|\nabla f(x_*)\|_*\|x - x_*\| + 1/2\, L\|x - x_*\|^2;$$

and hence,

$$\min_{k\in I} f(x^k) - f(x_*) \le \min_{k\in I}\left\{\|\nabla f(x_*)\|_*\|x^k - x_*\| + \frac{1}{2}\, L\|x^k - x_*\|^2\right\},$$

$$\min_{k\in I} f(x^k) - f(x_*) \le \|\nabla f(x_*)\|_*\frac{\varepsilon}{M_g} + \frac{L}{2}\,\frac{\varepsilon^2}{M_g^2}.$$

The last inequality allows us to formulate the following statement for some class of generally nonsmooth objective functionals [18]:

Corollary 1. *Suppose that*

$$f(x) = \max_{i=\overline{1,m}} f_i(x),$$

where f_i are differentiable at each $x \in X$ and

$$\|\nabla f_i(x) - \nabla f_i(y)\|_* \le L_i\|x - y\| \quad \forall x, y \in X.$$

Then, after

$$N = \lceil 2M_g^2\Theta_0^2/\varepsilon^2\rceil$$

steps of Algorithm 2, the following holds:

$$\min_{k\in I} f(x^k) - f(x_*) \le \|\nabla f(x_*)\|_*\frac{\varepsilon}{M_g} + \frac{L}{2}\,\frac{\varepsilon^2}{M_g^2}, \qquad L = \max_{i=\overline{1,m}} L_i.$$

Remark 2. We consider conditional problems, and therefore it is not necessary that $\|\nabla f(x_*)\|_* = 0$.

We describe the estimates for the convergence rate of Algorithm 1 and Algorithm 2 for the classes of objective functionals which were not considered in [18].

Remark 3. Let an objective functional $f\colon X \to \mathbb{R}$ satisfy the Lipschitz condition

$$|f(x) - f(y)| \le M_f \|x - y\| \qquad \forall\, x, y \in X. \tag{17}$$

Then $f(x) \le f(x_*) + M_f \|x - x_*\|$ for arbitrary $x \in X$; and so,

$$\min_{k \in I} f(x^k) - f(x_*) \le \min_{k \in I} M_f \{\|x^k - x_*\|\}.$$

Therefore, we have the following

Corollary 2. *Let f satisfy the Lipschitz condition* (17) *on X. Then*

(1) *after*

$$N = \lceil 2 \max\{1, M_g^2\} \cdot \Theta_0^2 / \varepsilon^2 \rceil$$

steps of Algorithm 1*, the inequality*

$$\min_{k \in I} f(x^k) - f(x_*) \le M_f \varepsilon$$

is true;

(2) *after*

$$N = \lceil 2 M_g^2 \Theta_0^2 / \varepsilon^2 \rceil$$

steps of Algorithm 2*, the following holds:*

$$\min_{k \in I} f(x^k) - f(x_*) \le \frac{M_f}{M_g}\, \varepsilon.$$

Remark 4. Let the objective functional $f\colon X \to \mathbb{R}$ be twice differentiable on X, and let f have the Lipschitz Hessian; i.e.,

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_* \le L \|x - y\| \qquad \forall\, x, y \in X. \tag{18}$$

In this case, for every $x \in X$ the following holds (see [3, Lemma 1.2.4]):

$$\left| f(x) - f(x_*) - \langle \nabla f(x_*), x - x_* \rangle - \frac{1}{2} \langle \nabla^2 f(x_*)(x - x_*), x - x_* \rangle \right| \le \frac{L}{6} \|x - x_*\|^3,$$

whence,

$$f(x) \le f(x_*) + \|\nabla f(x_*)\| \cdot \|x - x_*\| + \frac{1}{2} \|\nabla^2 f(x_*)(x - x_*)\| \cdot \|x - x_*\| + \frac{L}{6} \|x - x_*\|^3.$$

Therefore, for an arbitrary $x \in X$ we have

$$f(x) \le f(x_*) + \|\nabla f(x_*)\| \cdot \|x - x_*\| + \frac{1}{2} \|\nabla^2 f(x_*)\|_{\text{Fro}} \cdot \|x - x_*\|^2 + \frac{L}{6} \|x - x_*\|^3,$$

where $\|A\|_{\text{Fro}} = \operatorname{tr}(A^\top A)$ is the Frobenius norm of the matrix $A \in \mathbb{R}^{m \times n}$. Then

$$\min_{k \in I} f(x^k) - f(x_*) \le \min_{k \in I} \biggl\{ \|\nabla f(x_*)\| \cdot \|x^k - x_*\| + \frac{1}{2} \|\nabla^2 f(x_*)\|_{\text{Fro}} \cdot \|x^k - x_*\|^2 + \frac{L}{6} \|x^k - x_*\|^3 \biggr\}.$$

Combining the statements of Theorem 1 and Lemma 2, we obtain

$$f(x) - f(x_*) \le \|\nabla f(x_*)\|_* \cdot \varepsilon + \frac{1}{2} \|\nabla^2 f(x_*)\|_{\text{Fro}} \cdot \varepsilon^2 + \frac{L}{6} \varepsilon^3,$$

and, by analogy, from Theorem 2 we have

$$f(x) - f(x_*) \le \|\nabla f(x_*)\|_* \cdot \frac{\varepsilon}{M_g} + \frac{1}{2} \|\nabla^2 f(x_*)\|_{\text{Fro}} \cdot \frac{\varepsilon^2}{M_g^2} + \frac{L}{6} \frac{\varepsilon^3}{M_g^3}.$$

Therefore, the following is true:

Corollary 3. *Let f be twice differentiable on X, and let f have a Lipschitz Hessian; and so,* (18) *is true. Then*

(1) *after*

$$N = \lceil 2\max\{1, M_g^2\}\,\Theta_0^2/\varepsilon^2 \rceil$$

steps of Algorithm 1*, we obtain*

$$\min_{k\in I} f(x^k) - f(x_*) \le \|\nabla f(x_*)\|_* \cdot \varepsilon + \frac{1}{2}\,\|\nabla^2 f(x_*)\|_{\mathrm{Fro}} \cdot \varepsilon^2 + \frac{L}{6}\,\varepsilon^3;$$

(2) *after*

$$N = \lceil 2M_g^2\Theta_0^2/\varepsilon^2 \rceil$$

steps of Algorithm 2*, the following holds:*

$$\min_{k\in I} f(x^k) - f(x_*) \le \|\nabla f(x_*)\|_*\,\frac{\varepsilon}{M_g} + \frac{1}{2}\,\|\nabla^2 f(x_*)\|_{\mathrm{Fro}} \cdot \frac{\varepsilon^2}{M_g^2} + \frac{L}{6}\,\frac{\varepsilon^3}{M_g^3}.$$

Similar estimates can be written for some class of problems with nonsmooth objective functionals:

Corollary 4. *Suppose that*

$$f(x) = \max_{i=\overline{1,m}} f_i(x),$$

where f_i is twice differentiable at each point $x \in X$ and

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_* \le L_i\|x - y\| \qquad \forall\, x, y \in X.$$

Then

(1) *after*

$$N = \lceil 2\max\{1, M_g^2\}\,\Theta_0^2/\varepsilon^2 \rceil$$

steps of Algorithm 1*, the following holds:*

$$\min_{k\in I} f(x^k) - f(x_*) \le \|\nabla f(x_*)\|_* \cdot \varepsilon + \frac{1}{2}\,\|\nabla^2 f(x_*)\|_{\mathrm{Fro}} \cdot \varepsilon^2 + \frac{L}{6}\,\varepsilon^3, \qquad L = \max_{i=\overline{1,m}} L_i;$$

(2) *after*

$$N = \lceil 2M_g^2\Theta_0^2/\varepsilon^2 \rceil$$

steps of Algorithm 2*, the following is true:*

$$\min_{k\in I} f(x^k) - f(x_*) \le \|\nabla f(x_*)\|_* \cdot \frac{\varepsilon}{M_g} + \frac{1}{2}\|\nabla^2 f(x_*)\|_{\mathrm{Fro}} \cdot \frac{\varepsilon^2}{M_g^2} + \frac{L}{6}\,\frac{\varepsilon^3}{M_g^3}, \qquad L = \max_{i=\overline{1,m}} L_i.$$

3. ON ACCELERATION OF THE METHODS OF MIRROR DESCENT FOR STRONGLY CONVEX PROBLEMS

In this section we consider the problem

$$f(x) \to \min, \qquad g(x) \le 0, \qquad x \in X, \tag{19}$$

on assuming (3) as well as strong convexity of f and g with the same parameter $\mu > 0$. Let us slightly modify the assumptions on the prox function and assume $d(x)$ to be bounded on the unit ball of the selected norm $\|\cdot\|$:

$$d(x) \le \Theta_0^2 \qquad \forall x \in X\colon\ \|x\| \le 1. \tag{20}$$

We also assume that for $x^0 \in X$ there is $R_0 > 0$ such that $\|x^0 - x_*\|^2 \le R_0^2$.

We will consider the methods for finding an *ε-solution* $\widehat{x}$ (19):

$$f(\widehat{x}) - f(x_*) \le \varepsilon, \qquad g(\widehat{x}) \le \varepsilon.$$

To construct some methods for solving problem (19) under given assumptions, we use the idea of restarts of Algorithms 1 and 2. Consider an auxiliary statement (for instance, see [21]):

Lemma 3. *Let f and g be μ-strongly convex functionals with respect to the norm $\|\cdot\|$ on X,*

$$x_* = \arg\min_{x\in X} f(x),$$

$g(x) \le 0$ for $x \in X$, and

$$f(x) - f(x_*) \le \varepsilon_f, \qquad g(x) \le \varepsilon_g \tag{21}$$

for some $\varepsilon_f > 0$ and $\varepsilon_g > 0$. Then

$$\frac{\mu}{2}\,\|x - x_*\|^2 \le \max\{\varepsilon_f,\ \varepsilon_g\}. \tag{22}$$

Suppose that

$$f(x) = \max_{i=\overline{1,m}} f_i(x),$$

where f_i are differentiable at each $x \in X$ and have Liptschitz gradient; i.e., there exist $L_i > 0$ such that

$$\|\nabla f_i(x) - \nabla f_i(y)\|_* \le L_i\|x - y\| \qquad \forall\, x, y \in X. \tag{23}$$

Consider the function $\tau\colon \mathbb{R}^+ \to \mathbb{R}^+$ such that

$$\tau(\delta) = \max\left\{\delta\|\nabla f(x_*)\|_* + \frac{\delta^2 L}{2},\ \delta\right\}, \qquad L := \max_{i=\overline{1,m}}\{L_i\}. \tag{24}$$

Clearly, τ increases and $\tau(0) = 0$; therefore, for each $\varepsilon > 0$ there exists

$$\widehat{\varphi}(\varepsilon) > 0, \qquad \tau(\widehat{\varphi}(\varepsilon)) = \varepsilon.$$

Now we propose an adaptive Algorithm 3 for problem (19):

Algorithm 3 (adaptive mirror descent algorithm for strongly convex functionals)

```
Input: accuracy $\varepsilon > 0$; initial point $x^0$; $\Theta_0$ such that $d(x) \le \Theta_0^2 \quad \forall x \in X\colon \|x\| \le 1$; $X$; $d(\cdot)$; the strong convexity parameter $\mu$; $R_0$ satisfies the estimate$\|x^0 - x_*\|^2 \le R_0^2$.
    Set $d_0(x) = d\left(\frac{x - x^0}{R_0}\right)$.
    Set $p = 1$.
repeat
    Set $R_p^2 = R_0^2 \cdot 2^{-p}$.
    Set $\varepsilon_p = \frac{\mu R_p^2}{2}$.
    Set $x^p$        ▷ output of Algorithm 1 with accuracy $\widehat{\varphi}(\varepsilon_p)$, prox function $d_{p-1}(\cdot)$, and $\Theta_0^2$
    $d_p(x) \leftarrow d\big(d\frac{x-x^p}{R_p}\big)$.
    Set $p = p + 1$.
until $p > \log_2 \frac{\mu R_0^2}{2\varepsilon}$
```

Theorem 3. *Suppose that f and g are functionals μ-strongly convex on $X \subset \mathbb{R}^n$, f has Liptschitz gradient satisfying* (23), *and $d(x) \leq \Theta_0^2$ for all $x \in X$ such that $\|x\| \leq 1$. Let the initial approximation $x^0 \in X$ and the number $R_0 > 0$ be given so that $\|x^0 - x_*\|^2 \leq R_0^2$. Then for $\widehat{p} = \lceil \log_2(\mu R_0^2)/(2\varepsilon) \rceil$ the output $x^{\widehat{p}}$ is an ε-solution to problem* (19) *and also*

$$\|x^{\widehat{p}} - x_*\|^2 \leq 2\varepsilon/\mu.$$

Wherein, the total number of iterations of Algorithm 1 *during implementation of Algorithm* 3 *does not exceed*

$$\widehat{p} + \sum_{p=1}^{\widehat{p}} \frac{2\Theta_0^2 \max\{1, M_g^2\}}{\widehat{\varphi}^2(\varepsilon_p)}, \qquad \text{where} \quad \varepsilon_p = \frac{\mu R_0^2}{2^{p+1}}.$$

Proof. The function $d_p(x) = d\big((x - x^p)/R_p\big)$ defined in Algorithm 3 is 1-strongly convex with respect to the norm $\|\cdot\|/R_p$ for all $p \geq 0$. Using the mathematical induction, it is possible to prove that

$$\|x^p - x_*\|^2 \leq R_p^2 \qquad \forall\, p \geq 0.$$

For $p = 0$ this statement is obvious in view of the choice of x^0 and R_0.

Assume that for some p we have $\|x^p - x_*\|^2 \leq R_p^2$. We prove that $\|x^{p+1} - x_*\|^2 \leq R_{p+1}^2$. Since $d_p(x_*) \leq \Theta_0^2$, by Theorem 1, on the $(p+1)$th restart after at most N_{p+1} iterations of Algorithm 1, where

$$N_{p+1} = \left\lceil \frac{2\Theta_0^2 \max\{1, M_g^2\}}{\widehat{\varphi}^2(\varepsilon_{p+1})} \right\rceil,$$

we arrive at the following inequalities for $x^{p+1} = \bar{x}^{N_{p+1}}$:

$$f(x^{p+1}) - f(x_*) \leq \varepsilon_{p+1}, \qquad g(x^{p+1}) \leq \varepsilon_{p+1} \qquad \text{for} \quad \varepsilon_{p+1} = \mu R_{p+1}^2/2.$$

Then, by Lemma 3,

$$\|x^{p+1} - x_*\|^2 \leq 2\varepsilon_{p+1}/\mu = R_{p+1}^2.$$

So, for every $p \geq 0$ it is proved that

$$\|x^p - x_*\|^2 \leq R_p^2 = \frac{R_0^2}{2^p}, \qquad f(x^p) - f(x_*) \leq \frac{\mu R_0^2}{2^{p+1}}, \qquad g(x^p) \leq \frac{\mu R_0^2}{2^{p+1}}.$$

Therefore, when $p = \widehat{p} = \lceil \log_2(\mu R_0^2)/(2\varepsilon) \rceil$, the output x^p is the ε-solution of problem (19) and the following are true:

$$\|x^p - x_*\|^2 \leq R_p^2 = R_0^2/2^p \leq 2\varepsilon/\mu.$$

Suppose that K is the total number of iterations of Algorithm 1 during the operation of Algorithm 3 according to item 6 of the listing, whereas N_p is the total number of iterations of Algorithm 1 on the pth restart. Recall that the function $\tau\colon \mathbb{R}^+ \to \mathbb{R}^+$ increases, and for each $\varepsilon > 0$ there exists $\widehat{\varphi}(\varepsilon) > 0$ such that $\tau(\widehat{\varphi}(\varepsilon)) = \varepsilon$. Thus,

$$K = \sum_{p=1}^{\widehat{p}} N_p = \sum_{p=1}^{\widehat{p}} \left\lceil \frac{2\Theta_0^2 \max\{1, M_g^2\}}{\widehat{\varphi}^2(\varepsilon_p)} \right\rceil \leq \widehat{p} + \sum_{p=1}^{\widehat{p}} \frac{2\Theta_0^2 \max\{1, M_g^2\}}{\widehat{\varphi}^2(\varepsilon_p)}.$$

Theorem 3 is proved. □

Remark 5. The previous estimate for the number of iterations of Algorithm 1 can be somewhat specified in the case of $\varepsilon < 1$. In this case, for every $\delta < 1$ we have $\tau(\delta) \leq C\delta$ for some constant C.

Therefore, we can assume that $\widehat{\varphi}(\varepsilon) = \widehat{C} \cdot \varepsilon$ for the corresponding constant $\widehat{C} > 0$. It means that, on the $(p+1)$th restart of Algorithm 1, after at most

$$k_{p+1} = \left\lceil \frac{2\Theta_0^2 \max\{1, M_g^2\} R_p^2}{\widehat{\mathrm{C}}^2 \varepsilon_{p+1}^2} \right\rceil \tag{25}$$

iterations of Algorithm 1, the output x^{p+1} is guaranteed to satisfy the inequality

$$f(x^{p+1}) - f(x_*) \le \widehat{C} \cdot \varepsilon_{p+1}, \qquad g(x^{p+1}) \le \varepsilon_{p+1},$$

where $\varepsilon_{p+1} = \mu R_{p+1}^2/2$. Then, by Lemma 3,

$$\|x^{p+1} - x_*\|^2 \le \frac{2\max\{1, \widehat{C}\}\varepsilon_{p+1}}{\mu} = \max\{1, \widehat{C}\} \cdot R_{p+1}^2.$$

Thus, for all $p \ge 0$

$$\|x^p - x_*\|^2 \le \max\{1, \widehat{C}\} \cdot R_p^2 = \max\{1, \widehat{C}\} \cdot R_0^2 \cdot 2^{-p}.$$

At the same time, for every $p \ge 1$ the inequalities

$$f(x^p) - f(x_*) \le \max\{1, \widehat{C}\} \cdot \frac{\mu R_0^2}{2} \cdot 2^{-p}, \qquad g(x_p) \le \max\{1, \widehat{C}\} \cdot \frac{\mu R_0^2}{2} \cdot 2^{-p}.$$

hold. Thus, $p > \log_2((\mu R_0^2)/(2\varepsilon))$, then x^p will be the $(\max\{1, \widehat{C}\}\varepsilon)$-solution of the problem under consideration, wherein

$$\|x^p - x_*\|^2 \le \max\{1, \widehat{C}\} \cdot R_0^2 \cdot 2^{-p} \le \frac{2\varepsilon}{\mu}.$$

Let us estimate the total number N of iterations of Algorithm 1. Let $\widehat{p} = \lceil \log_2((\mu R_0^2)/(2\varepsilon)) \rceil$. Then, according to (25), up to multiplication by a constant, we have

$$N = \sum_{p=1}^{\widehat{p}} k_p \le \sum_{p=1}^{\widehat{p}} \left(1 + \frac{2\Theta_0^2 \max\{1, M_g^2\} R_p^2}{\varepsilon_{p+1}^2}\right) = \sum_{p=1}^{\widehat{p}} \left(1 + \frac{32\Theta_0^2 \max\{1, M_g^2\} 2^p}{\mu^2 R_0^2}\right)$$
$$\le \widehat{p} + \frac{64\Theta_0^2 \max\{1, M_g^2\} 2^{\widehat{p}}}{\mu^2 R_0^2} \le \widehat{p} + \frac{64\Theta_0^2 \max\{1, M_g^2\}}{\mu\varepsilon}.$$

Consider also the partially adaptive version of Algorithm 3 for problem (19) proposed in [18]:

Algorithm 4 (partially adaptive mirror descent algorithm for strongly convex functionals)

```
Input: accuracy $\varepsilon > 0$; initial point $x^0$; $\Theta_0$ such that $d(x) \le \Theta_0^2 \quad \forall x \in X\colon \ \|x\| \le 1$; $X$; $d(\cdot)$; the strong convexity parameter $\mu$; $R_0$ satisfies the estimate $\|x^0 - x_*\|^2 \le R_0^2$.
Set $d_0(x) = d\big((x - x^0)/R_0\big)$
Set $p = 1$.
repeat
    Set $R_p^2 = R_0^2 \cdot 2^{-p}$.
    Set $\varepsilon_p = \mu R_p^2/2$.
    Set $x^p$        $\triangleright$ output of Algorithm 2 with the accuracy $\varphi(\varepsilon_p)$, prox function $d_{p-1}(\cdot)$, and $\Theta_0^2$.
    $d_p(x) \leftarrow d\big((x - x^p)/R_p\big)$
    Set $p = p + 1$.
until $p > \log_2\big(\mu R_0^2/(2\varepsilon)\big)$
```

Under the conditions of Corollary 1, after stopping of Algorithm 4, the inequalities (21) will be satisfied for

$$\varepsilon_f = \frac{\varepsilon}{M_g}\|\nabla f(x_*)\|_* + \frac{\varepsilon^2 L}{2M_g^2}, \qquad \varepsilon_g = \varepsilon;$$

Consider the function $\tau\colon \mathbb{R}^+ \to \mathbb{R}^+$:

$$\tau(\delta) = \max\left\{\delta\|\nabla f(x_*)\|_* + \frac{\delta^2 L}{2};\ \delta M_g\right\}.$$

It is clear that τ increases, $\tau(0) = 0$, so, given $\varepsilon > 0$, there exists $\varphi(\varepsilon) > 0$ such that $\tau(\varphi(\varepsilon)) = \varepsilon$.

Theorem 4 [18]. *Let f and g be functionals μ-strongly convex on $X \subset \mathbb{R}^n$ and satisfying the conditions of Corollary 1, and let $d(x) \le \Theta_0^2$ for all $x \in X$ such that $\|x\| \le 1$. Suppose that the initial approximation $x^0 \in X$ and the number $R_0 > 0$ are given so that $\|x^0 - x_*\|^2 \le R_0^2$. Then for*

$$\widehat{p} = \left\lceil \log_2 \frac{\mu R_0^2}{2\varepsilon} \right\rceil$$

the output $x^{\widehat{p}}$ is an ε-solution of problem (19) *and*

$$\|x^{\widehat{p}} - x_*\|^2 \le \frac{2\varepsilon}{\mu}.$$

Wherein, the total number of iterations of Algorithm 2 during the implementation of Algorithm 4 does not exceed

$$\widehat{p} + \sum_{p=1}^{\widehat{p}} \frac{2\Theta_0^2 M_g^2}{\varphi^2(\varepsilon_p)}, \qquad \varepsilon_p = \frac{\mu R_0^2}{2^{p+1}}.$$

Remark 6. Generally speaking, $\varphi(\varepsilon)$ depends on $\|\nabla f(x_*)\|_*$ and the Lipschitz constant L for ∇f. If $\|\nabla f(x_*)\|_* < M_g$ then $\varphi(\varepsilon) = \varepsilon$ for sufficiently small ε:

$$\varepsilon < \frac{2(M_g - \|\nabla f(x_*)\|_*)}{L}.$$

For the other case ($\|\nabla f(x_*)\|_* > M_g$), for all $\varepsilon > 0$ we have

$$\varphi(\varepsilon) = \frac{\sqrt{\|\nabla f(x_*)\|_*^2 + 2\varepsilon L} - \|\nabla f(x_*)\|_*}{L}.$$

Remark 7. By analogy with the reasoning in Remark 5 for $\varepsilon < 1$, we can specify the upper estimate for the number of iterations of Algorithm 5 up to multiplication by a constant:

$$N = \widehat{p} + \frac{64\Theta_0^2 M_g^2 \cdot 2^{\widehat{p}}}{\mu^2 R_0^2} \le \widehat{p} + \frac{64\Theta_0^2 \cdot M_g^2}{\mu\varepsilon}.$$

Remark 8. By Corollaries 2 and 4, under the condition $\varepsilon < 1$, the statements of Remarks 5 and 7 can be easily transferred to the cases where the objective functional f satisfies the Lipschitz condition or the Lipschitz condition for its Hessian.

4. NUMERICAL EXPERIMENTS

4.1. Comparison of the Operation Speed of the Methods for the Fermat–Torricelli-Steiner Problem with Constraints

Note that in [14, Section 3.1] the following adaptive method is also proposed, which is optimal from the standpoint of lower oracle estimates in the case of problems with a Lipschitz objective functional:

Algorithm 5 (adaptive mirror descent (Lipschitz objective functional))

```
**Input:** $\varepsilon > 0$; $\Theta_0$ such that $d(x_*) \le \Theta_0^2$.
**Output:** $\bar{x}^N := \dfrac{\sum_{k\in I} x^k h_k}{\sum_{k\in I} h_k}$
$x^0 = \arg\min_{x\in X} d(x)$
$I := \varnothing$
$N \leftarrow 0$
**repeat**
    **if** $g(x^N) \le \varepsilon$ **then**
        $M_N = \|\nabla f(x^N)\|_*$, $h_N = \dfrac{\varepsilon}{M_N^2}$
        $x^{N+1} = \mathrm{Mirr}_{x^N}(h_N \nabla f(x^N))$            ▷ “productive steps”
        $N \to I$
    **else**
        $M_N = \|\nabla g(x^N)\|_*$, $h_N = \dfrac{\varepsilon}{M_N^2}$
        $x^{N+1} = \mathrm{Mirr}_{x^N}(h_N \nabla g(x^N))$            ▷ “unproductive steps”
    **endif**
    $N \leftarrow N + 1$
**until** $\sum_{j=0}^{N-1} \dfrac{1}{M_j^2} \ge 2\dfrac{\Theta_0^2}{\varepsilon^2}$
```

In the present work, an alternative method is considered (Algorithm 1) whose optimality can be established for conditional problems with a wider class of objective functionals (having a Lipschitz gradient or a Lipschitz Hessian). But it turns out that in the case of a Lipschitz objective functional, when we apply Algorithm 5, Algorithm 1 can work faster.

As an example, we present calculations for the well-known Fermat-Torricelli–Steiner problem with constraints:

Problem. *Given points $A_k = (a_{1k}, a_{2k}, \ldots, a_{10k})$ in the 10-dimensional Euclidean space $\mathbb{R}^{10}$, find the point $x = (x_1, \ldots, x_{10})$ such that the objective function*

$$f(x) := \sum_{k=1}^{10} \sqrt{(x_1 - a_{1k})^2 + (x_2 - a_{2k})^2 + \cdots + (x_{10} - a_{10k})^2}$$

assumes the minimal value on the set X which is determined by the constraints

$$g_1(x_1, \ldots, x_{10}) = 2x_1^2 + x_2^2 + \cdots + x_{10}^2 - 1 \le 0,$$

$$g_2(x_1, \ldots, x_{10}) = x_1^2 + 2x_2^2 + \cdots + x_{10}^2 - 1 \le 0,$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

$$g_{10}(x_1, \ldots, x_{10}) = x_1^2 + x_2^2 + \cdots + 2x_{10}^2 - 1 \le 0.$$

Table 1. Comparison of Algorithms 1, 5, and 6

ε	Iterations	Time, s	Iterations	Time, s	Iterations	Time, s
	Algorithm 5		Algorithm 1		Algorithm 6	
1/2	1659	97	283	15	231	6
1/4	5951	336	899	49	774	22
1/8	22356	1491	3159	180	2850	100

For $n = 10$ we give an example of the initial approximation $x^0 = (1, 1, \dots, 1)$ with the parameter $\Theta_0 = 3$ when choosing a standard Euclidean prox structure. The coordinates of points $A_k = (a_{1k}, a_{2k}, \dots, a_{10k})$ for $k = 1, 2, \dots, 10$ are chosen as rows of the following matrix A:

$$A = \begin{pmatrix} 1 & 2 & 1 & 4 & 1 & 0 & 4 & 4 & 4 & 3 \\ 2 & 4 & 3 & 1 & 0 & 2 & 4 & 0 & 4 & 0 \\ 3 & 2 & 3 & 4 & 3 & 0 & 3 & 4 & 2 & 3 \\ 0 & 0 & 2 & 0 & 2 & 4 & 4 & 1 & 0 & 0 \\ 3 & 3 & 4 & 4 & 3 & 0 & 1 & 0 & 4 & 4 \\ 2 & 2 & 4 & 0 & 4 & 0 & 2 & 2 & 1 & 1 \\ 0 & 4 & 3 & 4 & 2 & 3 & 3 & 4 & 0 & 2 \\ 2 & 2 & 1 & 4 & 2 & 1 & 4 & 3 & 0 & 3 \\ 4 & 1 & 2 & 2 & 3 & 3 & 2 & 1 & 3 & 1 \\ 3 & 3 & 2 & 2 & 0 & 0 & 4 & 0 & 3 & 4 \end{pmatrix}.$$

Note also that it is possible to accelerate the work of Algorithm 1 in case of several constraints due to the possibility of choosing a suitable constraint for unproductive iterations (see Algorithm 6 proposed in [15]). The corresponding results are shown in Table 1.

In Table 2 we present a comparison of the operation speed of the methods for the same parameters, but with nonsmooth functional constraints:

$$\begin{gathered} g_1(x_1, \dots, x_{10}) = 2|x_1| + |x_2| + \dots + |x_{10}| - 1 \leq 0, \\ g_2(x_1, \dots, x_{10}) = |x_1| + 3|x_2| + \dots + |x_{10}| - 1 \leq 0, \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ g_{10}(x_1, \dots, x_{10}) = |x_1| + |x_2| + \dots + 11|x_{10}| - 1 \leq 0. \end{gathered}$$

Table 2. Comparison of Algorithms 1, 5, and 6

ε	Iterations	Time, s	Iterations	Time, s	Iterations	Time, s
	Algorithm 5		Algorithm 1		Algorithm 6	
1/2	3709	279	671	29	437	21
1/4	14212	833	2418	103	1970	95
1/8	54655	2980	8979	455	8329	344

Algorithm 6 (modified adaptive mirror descent)

```
Input: $\varepsilon > 0$; $\Theta_0$ such that $d(x_*) \leq \Theta_0^2$.
Output: $\bar{x}^N := \arg \min\limits_{x^k, k \in I} f(x^k)$.
$x^0 = \arg\min\limits_{x \in X} d(x)$
$I := \varnothing$
$N \leftarrow 0$
repeat
    if $g(x^N) \leq \varepsilon$ then
        $h_N \leftarrow \dfrac{\varepsilon}{\|\nabla f(x^N)\|_*}$
        $x^{N+1} \leftarrow \mathrm{Mirr}_{x^N}(h_N \nabla f(x^N))$        ▷ “productive steps”
        $N \rightarrow I$
    else        ▷ $(g_{m(N)}(x^N) > \varepsilon)$ for some $m(N) \in \{1, \dots, M\}$
        $h_N \leftarrow \dfrac{\varepsilon}{\|\nabla g_{m(N)}(x^N)\|_*^2}$
        $x^{N+1} \leftarrow \mathrm{Mirr}_{x^N}(h_N \nabla g_{m(N)}(x^N))$        ▷ “unproductive steps”
    endif
    $N \leftarrow N + 1$
until $\Theta_0^2 \leq \dfrac{\varepsilon^2}{2}\Big(|I| + \sum\limits_{k \notin I} \frac{1}{\|\nabla g_{m(k)}(x^k)\|_*^2}\Big)$
```

4.2. On the Advantages of Using the Method with Restarts in the Strongly Convex Case

To demonstrate the advantages of Algorithm 3 as compared to Algorithm 1, a few numerical experiments were performed. Consider various 1-strongly convex objective functionals f, which have a Lipschitz gradient:

Example 1.

$$f(x) = \frac{L-\mu}{4}\left\{\frac{1}{2}\left[x_1^2 + \sum_{i=1}^{n-1}(x_i - x_{i+1})^2\right] - x_1\right\} + \frac{\mu}{2}\|x\|^2,$$

where $\mu = 1$, $L = 10\,000$, and $n = 10$.

Example 2. $f(x) = \max\{f_1(x), f_2(x), f_3(x)\}$, where

$$f_1(x) = \frac{1}{2}\big(x_1^2 + x_2^2 + 2x_3^2 + 4x_4^2 + x_5^2 + 5x_6^2 + 3x_7^2 + 2x_8^2 + 4x_9^2 + 8x_{10}^2\big) - \sum_{i=1}^{10} i x_i + 5,$$

$$f_2(x) = \frac{1}{2}\big(2x_1^2 + x_2^2 + 3x_3^2 + 4x_4^2 + 2x_5^2 + 5x_6^2 + x_7^2 + 6x_8^2 + 7x_9^2 + 2x_{10}^2\big) - \sum_{i=1}^{10}(10+i)x_i + 6,$$

$$f_3(x) = \frac{1}{2}\big(x_1^2 + x_2^2 + 2x_3^2 + 3x_4^2 + 5x_5^2 + x_6^2 + 4x_7^2 + 2x_8^2 + 3x_9^2 + 6x_{10}^2\big) - \sum_{i=1}^{10}(20+i)x_i + 7.$$

Example 3 (regression problem [22]).

$$f(x) = \frac{1}{2}\|Ax - b\|^2 + \frac{\mu}{2}\|x\|^2, \qquad \text{where}$$

$$A = \begin{pmatrix} 5 & 3 & 3 & 5 & 4 & 4 & 3 & 3 & 5 & 1 \\ 2 & 4 & 3 & 5 & 3 & 4 & 2 & 2 & 5 & 4 \\ 5 & 2 & 1 & 4 & 1 & 1 & 2 & 3 & 5 & 5 \end{pmatrix},$$

for $b = (1, 2, 3)^\top$ and $\mu = 1$.

Example 4 [22].

$$f(x) = \sum_{i=1}^{10} i x_i^4 + \frac{1}{2}\|x\|^2.$$

Example 5 [22]. The following test is carried out for a smoothed strongly convex version of the noise reduction problem

$$f(x) = \frac{1}{2}\|Ax - b\|^2 + \lambda\|x\|_{l_1,\tau} + \frac{\mu}{2}\|x\|^2,$$

$$A = \begin{pmatrix} 9 & 2 & 4 & 2 & 2 & 3 & 6 & 3 & 5 & 5 \\ 6 & 7 & 2 & 4 & 8 & 6 & 8 & 8 & 5 & 1 \end{pmatrix},$$

$$b = (1, 2)^\top, \qquad \mu = 1, \qquad \lambda = 0.05, \qquad \tau = 0.0001$$

and $\|\cdot\|_{l_1,\tau}$ is defined as follows:

$$\|x\|_{l_1,\tau} = \begin{cases} |x| - \frac{\tau}{2}, & \text{if} \quad |x| \geq \tau, \\ \frac{1}{2\tau}x^2, & \text{if} \quad |x| < \tau, \end{cases}$$

if x is a scalar and

$$\|x\|_{l_1,\tau} = \sum_{i=1}^{n} \|x_i\|_{l_1,\tau}$$

if $x = (x_1, x_2, \ldots, x_n)$ is a vector of $\mathbb{R}^n$.

Note that the quadratic term $\mu\|x\|^2/2$ guarantees strong convexity of the objective function.

Consider the functional constraints of the form $g(x) = G(x) + S(x)$ for

$$S(x) = \frac{1}{2}\|x\|^2, \qquad G(x) = \max_{i \in \overline{1,m}} g_i(x),$$

where $g_i(x) = \langle \alpha_i, x \rangle + \beta_i$, $\alpha_i^\top$ are the rows of the matrix

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 7 & 8 & 6 & 2 & 9 & 2 & 3 & 3 & 2 & 6 \\ 6 & 3 & 4 & 3 & 5 & 1 & 6 & 3 & 2 & 8 \\ 3 & 5 & 2 & 7 & 8 & 3 & 2 & 1 & 5 & 2 \\ 2 & 3 & 1 & 8 & 1 & 2 & 1 & 1 & 5 & 8 \\ 1 & 8 & 9 & 1 & 3 & 5 & 1 & 3 & 5 & 2 \\ 1 & 7 & 8 & 5 & 5 & 9 & 3 & 1 & 6 & 4 \\ 7 & 3 & 5 & 8 & 9 & 1 & 8 & 7 & 8 & 8 \\ 6 & 4 & 6 & 2 & 9 & 2 & 3 & 1 & 6 & 3 \\ 2 & 3 & 4 & 4 & 2 & 1 & 9 & 1 & 1 & 8 \end{pmatrix},$$

Table 3. Comparison of the results of work of Algorithms 1 and Algorithm 3

	Iterations	Time	Iterations	Time
	Algorithm 1		Algorithm 3	
Example 1	115 973	9:16	95 447	7:37
Example 2	57 798	7:01	45 455	5:14
Example 3	56 874	5:02	50 747	4:18
Example 4	13 720	1:15	6 764	0:38
Example 5	64 324	6:04	55 073	4:52

whereas the constants β_i are zero.

We assume that there is the standard Euclidean distance and the corresponding prox structure, and

$$X = B_1(0) = \{x = (x_1, x_2, \dots, x_{10}) \in \mathbb{R}^{10} \mid x_1^2 + x_2^2 + \dots + x_{10}^2 \le 1\},$$

the initial approximation

$$x^0 = \frac{(1,1,\dots,1)}{\|(1,1,\dots,1)\|}, \qquad \Theta_0 = 3, \qquad R_0 = 2,$$

and the accuracy ε equals 0.05.

The results of implementing Algorithms 1 and Algorithm 3 are presented in Table 3, where the number of iterations and the running time (in minutes and seconds) of each of these algorithms are given.

All calculations were performed using the Python 3.4 software on a computer equipped with Intel(R) Core(TM) i7-8550U CPU @ 1.80 GHz, 1992 MHz, 4 Core(s), 8 Logical Processor(s). The computer's RAM was 8 GB.

It is seen from Table 3 that Algorithm 3 works faster than Algorithm 1.

ACKNOWLEDGEMENTS

The authors are very grateful to Yu. E. Nesterov, A. V. Gasnikov and P. E. Dvurechensky for fruitful discussions and comments.

FUNDING

F. S. Stonyakin (analysis of Algorithms 1 and 3) was supported by the Russian Foundation for Basic Research (project no. 18–31–00219 mol_a).

REFERENCES

1. A. Ben-Tal and A. Nemirovski, "Robust Truss Topology Design via Semidefinite Programming," SIAM J. Optim. **7** (4), 991–1016 (1997).
2. Yu. Nesterov and S. Shpirko, "Primal-Dual Subgradient Methods for Huge-Scale Linear Conic Problem," SIAM J. Optim. **24** (3), 1444–1457 (2014).
3. Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course* (Kluwer Acad. Publ., Massachusetts, 2004).
4. F. P. Vasil'ev, *Optimization Methods*, Vol. 1 (MTsNMO, Moscow, 2011) [in Russian].
5. F. P. Vasil'ev, *Optimization Methods*, Vol. 2 (MTsNMO, Moscow, 2011) [in Russian].
6. G. Lan, "Gradient Sliding for Composite Optimization," Math. Program. **159** (1-2), 201–235 (2016).
7. A. Beck and M. Teboulle, "Mirror Descent and Nonlinear Projected Subgradient Methods for Convex Optimization," Oper. Res. Lett. **31** (3), 167–175 (2003).

8. A. Nemirovskii and D. Yudin, "Efficient Methods for Large-Scale Convex Optimization Problems," Ekonomika i Matematicheskie Metody No. 2, 135–152 (1979).
9. A. Nemirovski and D. Yudin, *Problem Complexity and Method Efficiency in Optimization* (Nauka, Moscow, 1979; J. Wiley & Sons, New York, 1983).
10. N. Z. Shor, "Application of Generalized Gradient Descent in Block Programming," Kibernetika **3** (3), 53–55 (1967).
11. B. T. Polyak, "A General Method of Solving Extremum Problems," Soviet Math. Dokl. **8** (3), 593–597 (1967).
12. A. Beck, A. Ben-Tal, N. Guttmann-Beck, and L. Tetruashvili, "The CoMirror Algorithm for Solving Nonsmooth Constrained Convex Problems," Oper. Res. Lett. **38** (6), 493–498 (2010).
13. A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization* (SIAM, Philadelphia, 2001).
14. A. Bayandina, P. Dvurechensky, A. Gasnikov, F. Stonyakin, and A. Titov, "Mirror Descent and Convex Optimization Problems with Non-Smooth Inequality Constraints," in *Large-Scale and Distributed Optimization*: *Lecture Notes in Mathematics*, Vol. 2227 (Springer, Cham, 2018), pp. 181–213.
15. F. S. Stonyakin, M. S. Alkousa, A. N. Stepanov, and M. A. Barinov, "Adaptive Mirror Descent Algorithms in Convex Programming Problems with Lipschitz Constraints," Trudy Inst. Mat. Mekh. Ural. Otdel. Ross. Akad. Nauk **24** (2), 266–279 (2018).
16. A. A. Titov, F. S. Stonyakin, A. V. Gasnikov, and M. S. Alkousa, "Mirror Descent and Constrained Online Optimization Problems Optimization and Applications," in *Optimization and Applications: Revised Selected Papers*, *9th International Conference OPTIMA-2018 (Petrovac, Montenegro, October 1-5, 2018)*. Ser. *Communications in Computer and Information Science*, Vol. 974 (Springer, Cham, 2019), pp. 64–78.
17. Yu. Nesterov, "Subgradient Methods for Convex Functions with Nonstandard Growth Properties," (2016). Available at `www.mathnet.ru:8080/PresentFiles/16179/growthbm_nesterov.pdf`.
18. F. S. Stonyakin and A. A. Titov, "One Mirror Descent Algorithm for Convex Constrained Optimization Problems with Non-Standard Growth Properties," in *Proceedings of the School-Seminar on Optimization Problems and Their Applications (OPTA-SCL), Omsk, Russia, July 8–14, 2018*, *CEUR Workshop Proceedings*, Vol. 2098 (RWTH Aachen Univ., Aachen, 2018), pp. 372–384. Available at `http://ceur-ws.org/Vol-2098`.
19. Yu. Nesterov, "A Method of Solving a Convex Programming Problem with Convergence Rate $O(1/k^2)$," Soviet Math. Doklady **27** (2), 372–376 (1983).
20. A. Juditsky and A. Nemirovski, "First Order Methods for Non-Smooth Convex Large-Scale Optimization. I: General Purpose Methods," in *Optimization for Machine Learning*, Ed. by S. Sra, S. Nowozin, and S. J. Wright (MIT Press, Cambridge, Massachusetts, 2012), pp. 121–148.
21. A. S. Bayandina, A. V. Gasnikov, E. V. Gasnikova, and S. V. Matsievsky, "Primal-Dual Mirror Descent for the Stochastic Programming Problems with Functional Constraints," Comput. Math. Math. Phys. **58** (11), 1728–1736 (2018).
22. X. Meng and H. Chen, "Accelerating Nesterov's Method for Strongly Convex Functions with Lipschitz Gradient," (Cornell Univ. Libr. e-Print Archive, 2011; arXiv: 1109.6058) `https://arxiv.org/pdf/1109.6058.pdf`.