КОМПЬЮТЕРНЫЕ ИССЛЕДОВАНИЯ
И МОДЕЛИРОВАНИЕ 2019 Т. 11 № 2 С. 205–217
DOI: 10.20537/2076-7633-2019-11-2-205-217

Ки&М

МАТЕМАТИЧЕСКИЕ ОСНОВЫ И ЧИСЛЕННЫЕ МЕТОДЫ МОДЕЛИРОВАНИЯ

УДК: 519.85

О некоторых стохастических методах зеркального спуска для условных задач онлайн-оптимизации

М. С. Алкуса

Московский физико-технический институт (государственный университет),
141701, Московская область, г. Долгопрудный, Институтский пер., 9

E-mail: mohammad.alkousa@phystech.edu

Получено 18.11.2018, после доработки — 05.03.2019.
Принято к публикации 06.03.2019.

Задача выпуклой онлайн-оптимизации естественно возникают в случаях, когда имеет место обновления статистической информации. Для задач негладкой оптимизации хорошо известен метод зеркального спуска. Зеркальный спуск — это расширение субградиентного метода для решения негладких выпуклых задач оптимизации на случай неевкидова расстояния. Работа посвящена стохастическим аналогам недавно предложенных методов зеркального спуска для задач выпуклой онлайн-оптимизации с выпуклыми липшицевыми (вообще говоря, негладкими) функциональными ограничениями. Это означает, что вместо (суб)градиента целевого функционала и функционального ограничения мы используем их стохастические (суб)градиенты. Точнее говоря, допустим, что на замкнутом подмножестве n-мерного векторного пространства задано N выпуклых липшицевых негладких функционалов. Рассматривается задача минимизации среднего арифметического этих функционалов с выпуклым липшицевым ограничением. Предложены два метода для решения этой задачи с использованием стохастических (суб)градиентов: адаптивный (не требует знания констант Липшица ни для целевого функционала, ни для ограничения), а также неадаптивный (требует знания константы Липшица для целевого функционала и ограничения). Отметим, что разрешено вычислять стохастический (суб)градиент каждого целевого функционала только один раз. В случае неотрицательного регрета мы находим, что количество непродуктивных шагов равно $O(N)$, что указывает на оптимальность предложенных методов. Мы рассматриваем произвольную прокс-структуру, что существенно для задач принятия решений. Приведены результаты численных экспериментов, позволяющие сравнить работу адаптивного и неадаптивного методов для некоторых примеров. Показано, что адаптивный метод может позволить существенно улучшить количество найденного решения.

Ключевые слова: задача выпуклой онлайн-оптимизации, негладкая задача условной оптимизации, адаптивный зеркальный спуск, липшицев функционал, стохастический (суб)градиент

COMPUTER RESEARCH AND MODELING
2019 VOL. 11 NO. 2 P. 205–217
DOI: 10.20537/2076-7633-2019-11-2-205-217

Ки&М

MATHEMATICAL MODELING AND NUMERICAL SIMULATION

UDC: 519.85

On some stochastic mirror descent methods for constrained online optimization problems

M. S. Alkousa

Moscow Institute of Physics and Technology,
9 Institutskiy per., Dolgoprudny, Moscow Region, 141701, Russia

E-mail: mohammad.alkousa@phystech.edu

Received 18.11.2018, after completion — 05.03.2019.
Accepted for publication 06.03.2019.

The problem of online convex optimization naturally occurs in cases when there is an update of statistical information. The mirror descent method is well known for non-smooth optimization problems. Mirror descent is an extension of the subgradient method for solving non-smooth convex optimization problems in the case of a non-Euclidean distance. This paper is devoted to a stochastic variant of recently proposed Mirror Descent methods for convex online optimization problems with convex Lipschitz (generally, non-smooth) functional constraints. This means that we can still use the value of the functional constraint, but instead of (sub)gradient of the objective functional and the functional constraint, we use their stochastic (sub)gradients. More precisely, assume that on a closed subset of n-dimensional vector space, N convex Lipschitz non-smooth functionals are given. The problem is to minimize the arithmetic mean of these functionals with a convex Lipschitz constraint. Two methods are proposed, for solving this problem, using stochastic (sub)gradients: adaptive method (does not require knowledge of Lipschitz constant neither for the objective functional, nor for the functional of constraint) and non-adaptive method (requires knowledge of Lipschitz constant for the objective functional and the functional of constraint). Note that it is allowed to calculate the stochastic (sub)gradient of each functional only once. In the case of non-negative regret, we find that the number of non-productive steps is $O(N)$, which indicates the optimality of the proposed methods. We consider an arbitrary proximal structure, which is essential for decision-making problems. The results of numerical experiments are presented, allowing to compare the work of adaptive and non-adaptive methods for some examples. It is shown that the adaptive method can significantly improve the number of the found solutions.

Keywords: online convex optimization problem, non-smooth constrained optimization problem, adaptive mirror Descent, Lipschitz functional, Stochastic (sub)gradient

Citation: *Computer Research and Modeling*, 2019, vol. 11, no. 2, pp. 205–217 (Russian).

1. Introduction

Via its powerful modeling capability for a lot of problems from diverse domains, online convex optimization (OCO) has become a leading online learning framework in recent years. For example, selection for search engines and spam filtering can all be modeled as special cases. OCO plays a key role in solving the problems where statistical information is being updated [Hazan, Kale, 2014; Hazan, 2015]. There are a lot of examples of such problems, concerning internet network, consumer data sets or financial market, and in machine learning applications such as adaptive routing in networks and online display advertising [Jenatton et al., 2015; Awerbuch, Kleinberg, 2008], online regression, online ranking, online shortest paths, and portfolio selection. See [Hazan, Kale, 2014; Hazan, 2015] for more applications and background. In OCO, the convex set is known in advance, but in each step of some repeated optimization problem, one must select a point in this convex set before seeing the objective function for that step. This can be used to model factory production, farm production, and many other industrial optimization problems where one is unaware of the value of the items produced until they have already been constructed [Zinkevich, 2003]. In an online decision problem, one has to make a sequence of decisions without knowledge of the future. The problem of prediction from expert advice is a special case of OCO in which the decision set is the unit simplex [Hazan, 2015; Kalai, Vempala, 2005]. In each period, we select one expert and then observe the cost lie on the unit simplex for each expert. Our goal is to minimize the arithmetic mean of costs from the point of view of all experts. In recent years, methods for solving online optimization problems have been actively developed [Bubeck, Eldan, 2015; Bubeck, Cesa-Bianchi, 2012; Gasnikov et al., 2015; Gasnikov et al., 2017; Hazan, Kale, 2014; Hazan, 2015; Jenatton et al., 2015; Lugosi, Cesa-Bianchi, 2006]. In [Titov et al., 2019] two methods (adaptive and non-adaptive) were proposed to solve the online optimization problem, with functional constraints, for an arbitrary prox-structure.

In problems of OCO, it is required to minimize the sum (or the arithmetic mean) of several convex Lipschitz functionals f_i ($i = \overline{1,N}$) given on some closed set $Q \subset \mathbb{R}^n$.

$$\begin{cases} \dfrac{1}{N}\displaystyle\sum_{i=1}^{N} f_i(x) \to \min\limits_{x\in Q}, \\ s.t.\ \ g(x) \le 0. \end{cases} \tag{1}$$

This paper is devoted to a stochastic variant of optimal adaptive and non-adaptive methods (see [Titov et al., 2019], Algorithms 1 and 2), for the randomized version of the type of problem (1). This means that we can still use the value of the function $g(x)$, but instead of (sub)gradient of f_i, $i = \overline{1,N}$ and g, we use their stochastic (sub)gradients $\nabla f_i(x,\xi), \nabla g(x,\zeta)$, where ξ, ζ are random vectors. It should be noted that it is possible to calculate the stochastic (sub)gradient of each functional f_i only once.

We assume that f_i and g are Lipschitz functionals, i.e. there exists a number $M > 0$, such that

$$\begin{gathered} |g(x) - g(y)| \le M\|x - y\|, \\ |f_i(x) - f_i(y)| \le M\|x - y\| \quad \forall i = \overline{1,N}. \end{gathered} \tag{2}$$

The optimization problems of non-smooth functionals with constraints frequently appear in huge-scale problems and their applications [Ben-Tal, Nemirovski, 1997; Shpirko, Nesterov, 2014]. For solving this kind of problems, there are several methods, such as bundle-level method [Nesterov, 2004], Lagrange multipliers method [Boyd, Vandenberghe, 2004] and Mirror Descent method, which originated in [Nemirovski, 1979; Nemirovsky, Yudin, 1983] and was later analyzed in [Beck, Teboulle, 2003]. Mirror Descent method is considered as the non-Euclidean extension of subgradient methods, which are considered in [Shor, 1967] for deterministic unconstrained problems and Euclidean setting,

and for constrained problems in [Polyak, 1967]. An extension of the Mirror Descent method for constrained problems was proposed in [Nemirovsky, Yudin, 1983; Beck et al., 2010].

Usually, the step size and stopping rule for Mirror Descent requires to know the Lipschitz constant of the objective function and constraint, if any. Adaptive step sizes, which do not require this information, are considered for unconstrained problems in [Ben-Tal, Nemirovski, 2001], and in [Beck et al., 2010] for constrained problems. In [Bayandina et al., 2018b] proposed some optimal Mirror Descent algorithms, for Lipschitz functional constraints problems with both adaptive step sizes and stopping rules. Also, there were considered some modifications of these methods for the case of problems with many functional constraints in [Stonyakin et al., 2018]. For OCO problem with constraints, in [Jenatton et al., 2015] authors considered adaptive algorithms, but with only standard Euclidean prox-structure. In [Hao et al., 2017] authors proposed an algorithm for OCO with stochastic constraints, where the objective functional varies arbitrarily but the constraint functionals are varying independently and identically distributed (i.i.d.) over time. In this paper, the objective functional and the constraint functionals are arbitrarily varying, but instead of calculating their (sub)gradient we calculate their stochastic (sub)gradient, which is very effective and requirable in huge-scale optimization problems.

In this paper, we propose adaptive and non-adaptive stochastic algorithms for solving the randomized version of the problem (1). We consider arbitrary proximal structure. The paper consists of an Introduction and four main sections. In Section 2 we give some basic notation concerning convex optimization problems with functional constraints and online optimization problems. In section 3 we propose a non-adaptive stochastic algorithm of Mirror Descent for the randomized considered online optimization problem (1). Section 4 is devoted to an adaptive analog of this method (Algorithm 2). In the last section, we consider some numerical experiments that allow us to compare the work of Algorithms 1 and 2 for certain examples.

2. Problem Statement and Standard Mirror Descent Basics

Let $(E, \|\cdot\|)$ be a normed finite-dimensional vector space and E^* be the conjugate space of E with the norm:

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

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

Let $Q \subset E$ be a (simple) closed convex set, $d : Q \to \mathbb{R}$ be a distance generating function (d.g.f.), which is continuously differentiable and 1-strongly convex with respect to the norm $\|\cdot\|$, i.e.

$$\forall x, y \in Q; \;\; d(y) \geq d(x) + \langle \nabla d(x), y - x\rangle + \frac{1}{2}\|y - x\|^2,$$

and assume that $\min\limits_{x\in Q} d(x) = d(0)$. Suppose, we have a constant Θ_0 such that $d(x_*) \leq \Theta_0^2$, where x_* is a solution of (1).

Note that if there is a set of optimal points for (1) $X_* \subset Q$, we may assume that

$$\min_{x_*\in X_*} d(x_*) \leq \Theta_0^2. \tag{3}$$

For all $x, y \in Q \subset E$ consider the corresponding Bregman divergence

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

We also assume that we know a constant $\Theta_0 > 0$ such that

$$\sup_{x,y\in Q} V_x(y) \leq \Theta_0^2. \tag{4}$$

Standard proximal setups, i.e. Euclidean, entropy, ℓ_1/ℓ_2, simplex, nuclear norm, spectahedron can be found, e.g. in [Ben-Tal, Nemirovski, 2001]. There are well-known examples of distance generating functions, let us denote ℓ_p norm by $\|x\|_p$, and the standard unit simplex in $\mathbb{R}^n$ by

$$S_n(1) = \left\{ x \in \mathbb{R}^n_+ \mid \sum_{i=1}^n x_i = 1 \right\}.$$

Consider two cases.

- **Entropy proximal function.** If $p = 1$, then over any $Q \subseteq S_n(1)$

$$d(x) = \sum_{k=1}^n x_k \ln x_k, \quad V_x(y) = \sum_{k=1}^n y_k \ln\left(\frac{y_k}{x_k}\right). \tag{5}$$

- **Standard Euclidean proximal function.** If $p = 2$, then over every Q

$$d(x) = \frac{1}{2}\|x\|_2^2, \quad V_x(y) = \frac{1}{2}\|x - y\|_2^2. \tag{6}$$

For all $x \in Q$, and $p \in E^*$, the proximal mapping operator (Mirror Descent step) is defined as

$$\mathrm{Mirr}_x(p) = \arg\min_{u \in Q} \{\langle p, u\rangle + V_x(u)\}.$$

We make the simplicity assumption, which means that $\mathrm{Mirr}_x(p)$ is easily computable.

Now, for the randomized version of the problem (1), we introduce the following assumptions (see [Bayandina et al., 2018b]). Given a point $x \in Q$, we can calculate the stochastic (sub)gradients $\nabla f_i(x, \xi)$, $i = \overline{1, N}$; $\nabla g(x, \zeta)$, where ξ, ζ are random vectors. These stochastic (sub)gradients satisfy

$$\mathbb{E}[\nabla f_i(x, \xi)] = \nabla f_i(x) \in \partial f_i(x) \ (i = \overline{1, N}), \qquad \mathbb{E}[\nabla g(x, \zeta)] = \nabla g(x) \in \partial g(x), \tag{7}$$

and

$$\|\nabla f_i(x, \xi)\|_* \le M \quad (i = \overline{1, N}), \qquad \|\nabla g(x, \zeta)\|_* \le M, \qquad a.s. \text{ in } \xi, \zeta. \tag{8}$$

To motivate these assumptions, we consider, in the standard unit simplex, the following problem (see [Bayandina et al., 2018b])

$$\begin{cases} f(x) = \dfrac{1}{2}\langle Ax, x\rangle \to \min\limits_{x \in S_n(1)}, \\ s.t. \ \ g(x) = \max\limits_{i=\overline{1,m}} \{\langle c_i, x\rangle\} \le 0, \end{cases}$$

where A is a given $n \times n$ matrix and c_i ($i = \overline{1, m}$) are given vectors in $\mathbb{R}^n$.

The exact computation of the gradient $\nabla f(x) = Ax$ takes $O(n^2)$ arithmetic operations, which is bad for huge-scale optimization problems, where n is large. In this setting, it is natural to use randomization to construct a stochastic approximation for $\nabla f(x)$. Let ξ be a random variable taking its values in $\{1, \ldots, n\}$ with probabilities $(x_1, \ldots, x_n)$ respectively. Let $A^{\langle i \rangle}$ denote the i-th column of the matrix A. Since $x \in S_n(1)$,

$$\mathbb{E}[A^{\langle \xi \rangle}] = A^{\langle i \rangle} \underbrace{\mathbb{P}(\xi = 1)}_{x_1} + \ldots + A^{\langle n \rangle} \underbrace{\mathbb{P}(\xi = n)}_{x_n} =$$
$$= A^{\langle 1 \rangle} x_1 + \ldots + A^{\langle n \rangle} x_n = Ax.$$

Thus, we can use $A^{\langle \xi \rangle}$ as a stochastic subgradient of f, which can be calculated in $O(n)$ arithmetic operations.

The following well-known lemma describes the main property of the proximal mapping operator (see, e.g. [Ben-Tal, Nemirovski, 2001; Bayandina et al., 2018b]).

Lemma 1. *Let $f\colon Q \to \mathbb{R}$ be a convex subdifferentiable function over the convex set Q and $z = Mirr_y(h\nabla f(y,\xi))$ for some $h > 0$, $y, z \in Q$ and ξ random vector. Then for each $x \in Q$*

$$h\,(f(y) - f(x)) \leq \frac{h^2}{2}\|\nabla f(y,\xi)\|_*^2 + V_y(x) - V_z(x) + h\langle \nabla f(y,\xi) - \nabla f(y), y - x\rangle.$$

3. Non-Adaptive Stochastic Algorithm for Constrained Online Optimization Problems

Let I, J denote the set of indexes of productive and non-productive steps, respectively. N_J denote the number of non-productive steps.

In this section, we consider the stochastic version of the Non-Adaptive Mirror Descent Algorithm 1 in [Titov et al., 2019], for the randomized version of the problem (1), with a constant step, which depends on the Lipschitz constant M. The proposed algorithm will work until there are exactly N productive steps and in each step the stochastic (sub)gradient of exactly one functional of the objectives is calculated. As a result, we get a (random) sequence $\{x^k\}_{k\in I}$ on productive steps, which can be considered as a solution to the randomized version of the problem (1), with accuracy δ (see (9)).

Denote

$$\delta_k = \begin{cases} \langle \nabla f_i(x^k,\xi^k) - \nabla f_i(x^k), x^k - x_*\rangle, \text{ if } k \in I \text{ and } i = \overline{1,N}, \\ \langle \nabla g(x^k,\zeta^k) - \nabla g(x^k), x^k - x_*\rangle, \text{ if } k \in J. \end{cases}$$

Algorithm 1. Non-Adaptive Stochastic Online Mirror Descent Algorithm

Require: $\varepsilon, M, N, \Theta_0, Q, d(\cdot), x^0$.

$i := 1,\ k := 0;$
repeat
  if $g(x^k) \leqslant \varepsilon$ **then**
    $h = \frac{\varepsilon}{M^2};$
    $x^{k+1} := Mirr_{x^k}(h\nabla f_i(x^k,\xi^k));$ "productive steps"
    $i := i + 1;$
    $k := k + 1;$
  else
    $h = \frac{\varepsilon}{M^2};$
    $x^{k+1} := Mirr_{x^k}(h\nabla g(x^k,\zeta^k));$ "non-productive steps"
    $k := k + 1;$
  end if
until $i = N + 1$
Guaranteed accuracy:

$$\delta := \frac{\varepsilon}{2} + \frac{M^2\Theta_0^2}{\varepsilon N} - \frac{\varepsilon N_J}{2N} \tag{9}$$

By Lemma 1, with $y = x^k, z = x^{k+1}, x = x_*$ and by (8), we have for all $k \in I$

$$f_i(x^k) - f_i(x_*) \leq \frac{h}{2}M^2 + \frac{V_{x^k}(x_*)}{h} - \frac{V_{x^{k+1}}(x_*)}{h} + \langle \nabla f_i(x^k,\xi^k) - \nabla f_i(x^k), x^k - x_*\rangle. \tag{10}$$

By the definition of stepsize h, we can rewrite (10) to get

$$f_i(x^k) - f_i(x_*) \leq \frac{\varepsilon}{2} + \frac{V_{x^k}(x_*)}{h} - \frac{V_{x^{k+1}}(x_*)}{h} + \langle \nabla f_i(x^k,\xi^k) - \nabla f_i(x^k), x^k - x_*\rangle \tag{11}$$

the same for all $k \in J$, we have

$$g(x^k) - g(x_*) \le \frac{\varepsilon}{2} + \frac{V_{x^k}(x_*)}{h} - \frac{V_{x^{k+1}}(x_*)}{h} + \langle \nabla g(x^k, \zeta^k) - \nabla g(x^k), x^k - x_* \rangle. \tag{12}$$

Taking summation, in each side of (11) and (12), over productive and non-productive steps, we get

$$\sum_{i=1}^{N} (f_i(x^k) - f_i(x_*)) + \sum_{k \in J} (g(x^k) - g(x_*)) \le \frac{\varepsilon}{2}(N + N_J) + \\ + \frac{M^2}{\varepsilon} \sum_{k=0}^{N+N_J-1} (V_{x^k}(x_*) - V_{x^{k+1}}(x_*)) + \sum_{k=0}^{N+N_J-1} \delta_k.$$

Using (4) and because for $k \in J$ we have $g(x^k) - g(x_*) \ge g(x^k) > \varepsilon$, we get

$$\sum_{i=1}^{N} (f_i(x^k) - f_i(x_*)) \le \frac{\varepsilon}{2} N + \frac{M^2 \Theta_0^2}{\varepsilon} - \frac{\varepsilon}{2} N_J + \sum_{k=0}^{N+N_J-1} \delta_k. \tag{13}$$

Dividing each side of (13) by N, using (9), and by taking the expectation, we obtain

$$\mathbb{E}\left[\frac{1}{N} \sum_{i=1}^{N} f_i(x^k)\right] - \min_{x \in Q} \frac{1}{N} \sum_{i=1}^{N} f_i(x) \le \delta + \sum_{k=0}^{N+N_J-1} \mathbb{E}\left[\frac{\delta_k}{N}\right].$$

But $\sum\limits_{k=0}^{N+N_J-1} \mathbb{E}\left[\frac{\delta_k}{N}\right] = 0$ (see [Bayandina, 2017]). Thus

$$\mathbb{E}\left[\frac{1}{N} \sum_{i=1}^{N} f_i(x^k)\right] - \frac{1}{N} \sum_{i=1}^{N} f_i(x_*) \le \delta. \tag{14}$$

From (14) we have

$$\mathbb{E}\left[\sum_{i=1}^{N} f_i(x^k)\right] - \sum_{i=1}^{N} f_i(x_*) \le N\delta = \frac{\varepsilon}{2} N + \frac{M^2 \Theta_0^2}{\varepsilon} - \frac{\varepsilon N_J}{2}. \tag{15}$$

If we assume the nonnegativity of the regret (i.e. the left side in (15)) and

$$\delta \le \varepsilon = \frac{C}{\sqrt{N}} \text{ for some } C > 0 \tag{16}$$

then we get

$$0 \le N + \frac{2M^2 \Theta_0^2}{\varepsilon^2} - N_J = N + \frac{2M^2 \Theta_0^2}{C^2} N - N_J,$$

then

$$N_J \le N \cdot \left(1 + \frac{2M^2 \Theta_0^2}{C^2}\right) \sim O(N).$$

Thus, we have the following result

Theorem 1. *Suppose Algorithm 1 works exactly N productive steps. After the stopping of the Algorithm 1, the following inequality holds:*

$$\mathbb{E}\left[\sum_{i=1}^{N} f_i(x^k)\right] - \sum_{i=1}^{N} f_i(x_*) \leq \delta.$$

For the case (16) *and*

$$\mathbb{E}\left[\sum_{i=1}^{N} f_i(x^k)\right] - \min_{x\in Q} \frac{1}{N} \sum_{i=1}^{N} f_i(x) \geq 0$$

there will be no more than

$$N \cdot \left(1 + \frac{2M^2\Theta_0^2}{C^2}\right) \sim O(N) \tag{17}$$

non-productive steps.

4. Adaptive Stochastic Algorithm for Constrained Online Optimization Problems

In this section, we consider the stochastic version of the Adaptive Mirror Descent Algorithm 2 in [Titov et al., 2019], for the randomized version of the problem (1). The proposed algorithm will work until there are exactly N productive steps. As a result, we get a (random) sequence $\{x^k\}_{k\in I}$ on productive steps, which can be considered as a solution to the randomized version of the problem (1), with accuracy δ (see (18)).

Algorithm 2. Adaptive Stochastic Online Mirror Descent Algorithm

Require: $\varepsilon, N, \Theta_0, Q, d(\cdot), x^0$.

$i := 1,\ k := 0;$
repeat
  if $g(x^k) \leqslant \varepsilon$ **then**
    $M_k := \|\nabla f_i(x^k, \xi^k)\|_*;$
    $h_k = \Theta_0 \left(\sum\limits_{t=0}^{k} M_t^2\right)^{-1/2};$
    $x^{k+1} := Mirr_{x^k}(h_k \nabla f_i(x^k, \xi^k));$ "productive steps"
    $i := i + 1;$
    $k := k + 1;$
  else
    $M_k := \|\nabla g(x^k, \zeta^k)\|_*;$
    $h_k = \Theta_0 \left(\sum\limits_{t=0}^{k} M_t^2\right)^{-1/2};$
    $x^{k+1} := Mirr_{x^k}(h_k \nabla g(x^k, \zeta^k));$ "non-productive steps"
    $k := k + 1;$
  end if
until $i = N + 1$
Guaranteed accuracy:

$$\delta := \frac{2\Theta_0}{N} \left(\sum_{i=0}^{N+N_J-1} M_i^2\right)^{1/2} - \varepsilon \cdot \frac{N_J}{N} \tag{18}$$

By Lemma 1, with $y = x^k, z = x^{k+1}, h = h_k, x = x_*$ we have for all $k \in I$

$$h_k\left(f_i(x^k) - f_i(x_*)\right) \le \frac{h_k^2}{2}\|\nabla f_i(x^k, \xi^k)\|_*^2 + V_{x^k}(x_*) - V_{x^{k+1}}(x_*) + h_k\langle \nabla f_i(x^k, \xi^k) - \nabla f_i(x^k), x^k - x_*\rangle \quad (19)$$

and, for all $k \in J$, we have

$$h_k\left(g(x^k) - g(x_*)\right) \le \frac{h_k^2}{2}\|\nabla g(x^k, \zeta^k)\|_*^2 + V_{x^k}(x_*) - V_{x^{k+1}}(x_*) + h_k\langle \nabla g(x^k, \zeta^k) - \nabla g(x^k), x^k - x_*\rangle. \quad (20)$$

Dividing each inequality, (19) and (20), by h_k and taking summation over productive and non-productive steps, we obtain

$$\sum_{i=1}^{N}\left(f_i(x^k) - f_i(x_*)\right) + \sum_{k\in J}\left(g(x^k) - g(x_*)\right) \le \sum_{k=0}^{N+N_J-1}\frac{h_k M_k^2}{2} + \sum_{k=0}^{N+N_J-1}\frac{1}{h_k}\left(V_{x^k}(x_*) - V_{x^{k+1}}(x_*)\right) + \sum_{k=0}^{N+N_J-1}\delta_k.$$

Using (4),

$$\sum_{k=0}^{N+N_J-1}\frac{1}{h_k}(V_{x^k}(x_*) - V_{x^{k+1}}(x_*)) = \frac{1}{h_0}V_{x^0}(x_*) + \sum_{k=0}^{N+N_J-2}\left[\Big(\frac{1}{h_{k+1}} - \frac{1}{h_k}\Big)V_{x^{k+1}}(x_*) - \frac{1}{h_{N+N_J-1}}V_{x^k}(x_*)\right] \le$$

$$\le \frac{\Theta_0^2}{h_0} + \Theta_0^2\sum_{k=0}^{N+N_J-2}\Big(\frac{1}{h_{k+1}} - \frac{1}{h_k}\Big) = \frac{\Theta_0^2}{h_{N+N_J-1}}.$$

Whence, by the definition of stepsizes h_k,

$$\sum_{i=1}^{N}\left(f_i(x^k) - f(x_*)\right) + \sum_{k\in J}\left(g(x^k) - g(x_*)\right) \le \sum_{k=0}^{N+N_J-1}\frac{\Theta_0}{2}\frac{M_k^2}{\left(\sum_{j=0}^{k}M_j^2\right)^{1/2}} + \Theta_0\left(\sum_{k=0}^{N+N_J-1}M_k^2\right)^{1/2} + \sum_{k=0}^{N+N_J-1}\delta_k \le$$

$$\le 2\Theta_0\left(\sum_{k=0}^{N+N_J-1}M_k^2\right)^{1/2} + \sum_{k=0}^{N+N_J-1}\delta_k$$

where we used the inequality

$$\sum_{k=0}^{N+N_J-1}\frac{M_k^2}{\left(\sum_{j=0}^{k}M_j^2\right)^{1/2}} \le 2\left(\sum_{k=0}^{N+N_J-1}M_k^2\right)^{1/2},$$

which can be proved by induction. Since, for $k \in J$, $g(x^k) - g(x_*) \ge g(x^k) > \varepsilon$, we get

$$\sum_{i=1}^{N}(f_i(x^k) - f_i(x^*)) < \varepsilon N - \varepsilon(N + N_J) + 2\Theta_0\left(\sum_{i=0}^{N+N_J-1}M_i^2\right)^{1/2} + \sum_{k=0}^{N+N_J-1}\delta_k. \quad (21)$$

Dividing each side of (21) by N, using (18) and by taking the expectation, we obtain

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}f_i(x^k)\right] - \min_{x\in Q}\frac{1}{N}\sum_{i=1}^{N}f_i(x) \le \delta + \sum_{k=0}^{N+N_J-1}\mathbb{E}\left[\frac{\delta_k}{N}\right].$$

Thus

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}f_i(x^k)\right] - \frac{1}{N}\sum_{i=1}^{N}f_i(x_*) \le \delta. \quad (22)$$

From (22) we have

$$\mathbb{E}\left[\sum_{i=1}^{N} f_i(x^k)\right] - \sum_{i=1}^{N} f_i(x_*) \leq N\delta = 2\Theta_0\left(\sum_{i=0}^{N+N_J-1} M_i^2\right)^{1/2} - \varepsilon(N_J + N) + \varepsilon N. \tag{23}$$

If we assume the nonnegativity of the regret (i.e. the left side in (23)) and the accuracy is given by (16), we can get

$$\varepsilon(N + N_J) \leq \varepsilon N + 2\Theta_0\left(\sum_{i=0}^{N+N_J-1} M_i^2\right)^{1/2} \leq \varepsilon N + 2M\Theta_0\sqrt{N + N_J},$$

$$N_J^2 \leq \frac{4M^2\Theta_0^2(N + N_J)}{\varepsilon^2} = \frac{4M^2\Theta_0^2(N + N_J)N}{C^2}.$$

Further,

$$\frac{N_J^2}{N^2 + NN_J} = \frac{\left(\frac{N_J}{N}\right)^2}{1 + \frac{N_J}{N}} \leq \frac{4M^2\Theta_0^2}{C^2}$$

and $N_J = O(N)$. Thus, we have come to the following result.

Theorem 2. *Suppose Algorithm 2 works exactly N productive steps. After the stopping of the Algorithm 2, the following inequality holds:*

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} f_i(x^k)\right] - \min_{x\in Q}\frac{1}{N}\sum_{i=1}^{N} f_i(x) \leq \delta.$$

For the case of (16) *and*

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} f_i(x^k)\right] - \min_{x\in Q}\frac{1}{N}\sum_{i=1}^{N} f_i(x) \geq 0$$

there will be no more than $O(N)$ non-productive steps.

Remark 1. In the case of negative regret (see [Titov et al., 2019]), i.e. the left side in (15) and (23) is negative, note that the set of productive steps is not empty, because for arbitrary p steps when the inequality $\sum_{k=1}^{p}\frac{1}{M_k^2} \geq \frac{2\Theta_0^2}{\varepsilon^2}$ is satisfied, one of these p steps will necessarily be productive. If all the other $p-1$ steps are non-productive (without loss of generality, let the last step be productive), then $\sum_{k=1}^{p-1}\frac{1}{M_k^2} < \frac{2\Theta_0^2}{\varepsilon^2}$, and $p - 1 < \frac{2M^2\Theta_0^2}{\varepsilon^2}$. Between each two successive productive steps there will be no more than $\frac{2M^2\Theta_0^2}{\varepsilon^2}$ non-productive steps, i.e. the number of all non-productive steps will be no more than $\frac{2M^2\Theta_0^2}{\varepsilon^2}N$. Therefore, as previous for $\varepsilon = \frac{C}{\sqrt{N}}$ there will be no more than

$$\frac{2M^2\Theta_0^2}{\varepsilon^2}N = O(N^2)$$

non-productive steps.

5. Numerical Experiments

To compare Algorithms 1 and 2, some numerical tests were carried out. Consider different examples with objective functional

$$f(x) = \frac{1}{N}\sum_{i=1}^{N}|\langle a_i, x\rangle - b_i|;\ a_i \in \mathbb{R}^{20},\ b_i \in \mathbb{R}.$$

Which is Lipschitz-continuous functional with constant $M_f = \frac{1}{N}\sum\limits_{i=1}^{N}\|a_i\|_2$. For the coefficients a_i and constants b_i for $i = 1, \ldots, N$, with different values of N. Let $A \in \mathbb{R}^{N\times 21}$ be a matrix with entries drawn from different random distributions. Then a_i^T are rows in the matrix $A' \in \mathbb{R}^{N\times 20}$, which is obtained from A, by eliminating the last column, and b_i are the entries of the last column in the matrix A. In details, entries of A are drawn:

- when $N = 10\,000$, from a normal (Gaussian) distribution with mean (center) equaling 0 and standard deviation (width) equaling 1;
- when $N = 20\,000$, from a uniform distribution over $[0, 1)$;
- when $N = 30\,000$, from the standard exponential distribution with a scale parameter of 1;
- when $N = 40\,000$, from a Gumbel distribution with the location of the mode equaling 1 and the scale parameter equaling 2;
- when $N = 50\,000$, from the discrete uniform distribution in the half open interval $[1, 11)$. These entries are integers in $[1, 10]$.

For the functional of constraints $g(x) = \max\limits_{i\in\overline{1,m}}\{g_i(x)\}$, we take $m = 10$ and the functionals $g_i(x) = \langle\alpha_i, x\rangle$, where α_i^T are the rows of the matrix

$$\begin{pmatrix}
-5 & -4 & -2 & 2 & 7 & 9 & 9 & 1 & -1 & 9 & -5 & -5 & -1 & 2 & 4 & -8 & 3 & -10 & -8 & 2\\
-4 & -3 & 2 & -2 & -3 & 5 & 0 & 8 & 2 & -7 & -3 & 2 & 5 & 4 & -7 & 7 & 9 & -7 & -10 & 4\\
3 & -3 & -4 & -10 & -1 & 8 & 7 & -6 & -4 & 4 & 6 & -6 & -10 & -5 & 5 & 8 & -1 & 9 & 7 & 7\\
-2 & 2 & -10 & 7 & 6 & -1 & -7 & 3 & -6 & -7 & -10 & -4 & 7 & 9 & -1 & -8 & -6 & -6 & -6 & 9\\
-8 & 3 & 0 & 3 & -6 & 2 & 2 & -6 & -3 & -7 & 6 & 6 & -10 & 5 & 2 & -10 & -10 & -4 & -5 & 1\\
-4 & 1 & -2 & -5 & -7 & 1 & 4 & -1 & 7 & 8 & -3 & 2 & -2 & 1 & 9 & 4 & 7 & -1 & 9 & -5\\
-6 & 7 & 4 & 9 & 1 & -10 & -2 & 6 & 7 & -1 & 3 & -8 & -9 & -3 & -1 & 8 & 3 & -7 & 9 & 4\\
3 & -9 & -6 & -2 & -5 & -2 & -7 & 1 & 4 & 8 & 1 & -5 & -1 & -6 & 5 & 3 & -10 & 9 & -10 & 9\\
-3 & -8 & -4 & 3 & -7 & -10 & -9 & 8 & -2 & 2 & 8 & -9 & -5 & 5 & 4 & -10 & -4 & -6 & 1 & 5\\
-2 & 1 & 6 & -2 & 2 & 1 & 4 & -8 & 9 & -1 & -10 & 6 & -4 & 4 & -10 & 2 & -7 & -4 & -7 & 8
\end{pmatrix}$$

The functional g is Lipschitz-continuous with constant $M_g = \max\limits_{i\in\overline{1,10}}\{\|\alpha_i\|_2\}$. In Algorithm 1 we take $M = \max\{M_f, M_g\}$. We choose the standard Euclidean proximal setup (see (6)), starting point $x^0 = \frac{(1, 1, \ldots, 1)}{\sqrt{20}}$, $\varepsilon = \frac{1}{\sqrt{N}}$, $Q = \{x = (x_1, x_2, \ldots, x_{20}) \in \mathbb{R}^{20} \,|\, x_1^2 + x_2^2 + \ldots + x_{20}^2 \le 1\}$. For any $x = (x_1, \ldots, x_{20})$ and $y = (y_1, \ldots, y_{20})$ in Q, the following inequality holds

$$\frac{1}{2}\|x - y\|_2^2 = \frac{1}{2}\sum_{k=1}^{20}(x_k - y_k)^2 \le x_1^2 + \ldots + x_{20}^2 + y_1^2 + \ldots + y_{20}^2 \le 2.$$

Therefore, we can choose $\Theta_0 = \sqrt{2}$.

The results of the work of Algorithms 1 and 2, are represented in Table 1 below. The number of non-productive steps is denoted by *nonprod.*, time is given in seconds and parts of the second, δ is guaranteed accuracy of the solution approximation found (sequence $\{x^k\}_{k\in I}$ on productive steps).

All experiments were implemented in Python 3.4, on a computer fitted with Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz, 1992 Mhz, 4 Core(s), 8 Logical Processor(s). RAM of the computer is 8 GB.

From Table 1 one can see, that the adaptive Algorithm 2 always works better than non-adaptive Algorithm 1. It is clearly shown in all experiments by the number of non-productive steps, the running time of the algorithms and guaranteed accuracy δ, where the number of the non-productive steps and δ obtained by Algorithm 2 are very small compared to the Algorithm 1.

Table 1. Results of Algorithms 1 and 2

	Algorithm 1			Algorithm 2		
N	nonprod.	time	δ	nonprod.	time	δ
10 000	30 866	05.685	16.729	392	01.026	0.042
20 000	43 810	08.137	11.833	248	01.894	0.018
30 000	54 005	10.898	9.662	943	03.064	0.028
40 000	63 171	13.238	8.368	2 759	04.667	0.051
50 000	71 757	14.970	7.485	4 398	05.615	0.080

6. Conclusions

In this work, two methods with the explicit form of steps, adaptive and non-adaptive, were proposed to solve the randomized version of the online optimization problem for an arbitrary proximal structure. The objective functional and the constraint functionals are arbitrarily varying, but instead of calculating their (sub)gradient we calculate their stochastic (sub)gradient, which is very effective and requirable in huge-scale optimization problems and their applications related to the Internet, machine learning and others. Furthermore, it has been proved that the number of non-productive steps is $O(N)$, in the case of non-negative regret. The future work, in connection with this work, implies considering a modification of the proposed adaptive algorithm for the case of a set of functional constraints, which make it possible to reduce the running time of the algorithm.

Acknowledgment. The author is very grateful to Alexander V. Gasnikov, Fedor S. Stonyakin and Alexander G. Biryukov for fruitful discussions.

Список литературы (References)

Awerbuch B., Kleinberg R. Online linear optimization and adaptive routing // Journal of Computer and System Sciences. — 2008. — Vol. 74, No. 1. — P. 97–114.

Bayandina A. Adaptive Stochastic Mirror Descent for Constrained Optimization // 2017. — https://arxiv.org/pdf/1705.02031.pdf

Bayandina A., Gasnikov A., Gasnikova E., Matsievsky S. Primal-dual mirror descent for the stochastic programming problems with functional constraints // Computational Mathematics and Mathematical Physics (accepted). — 2018a. — https://arxiv.org/pdf/1604.08194.pdf (in Russian).

Bayandina A., Dvurechensky P., Gasnikov A., Stonyakin F., Titov A. Mirror descent and convex optimization problems with non-smooth inequality constraints // Large-Scale and Distributed Optimization. Lecture Notes in Mathematics. — 2018b. — Vol. 2227. — P. 181–213.

Beck A., Ben-Tal A., Guttmann-Beck N., Tetruashvili L. The comirror algorithm for solving nonsmooth constrained convex problems // Operations Research Letters. — 2010. — Vol. 38, No. 6. — P. 493–498.

Beck A., Teboulle M. Mirror descent and nonlinear projected subgradient methods for convex optimization // Operations Research Letters. — 2003. — Vol. 31, No. 3. — P. 167–175.

Ben-Tal A., Nemirovski A. Lectures on Modern Convex Optimization. — Philadelphia: Society for Industrial and Applied Mathematics, 2001. — 590 p.

Ben-Tal A., Nemirovski A. Robust Truss Topology Design via semidefinite programming // SIAM Journal on Optimization. — 1997. — Vol. 7, No. 4. — P. 991–1016.

Boyd S., Vandenberghe L. Convex Optimization. — New York: Cambridge University Press, 2004. — 730 p.

Bubeck S., Eldan R. Multi-scale exploration of convex functions and bandit convex optimization // e-print — 2015. — http://research.microsoft.com/en-us/um/people/sebubeck/ConvexBandits.pdf

Bubeck S., Cesa-Bianchi N. Regret analysis of stochastic and nonstochastic multi-armed bandit problems // Foundation and Trends in Machine Learning. — 2012. — Vol. 5, No. 1. — P. 1–122.

Gasnikov A. V., Lagunovskaya A. A., Morozova L. E. On the relationship between simulation logit dynamics in the population game theory and a mirror descent method in the online optimization using the example of the shortest path problem // PROCEEDINGS OF MIPT. — 2015. — Vol. 7, No. 4. — P. 104–113 (in Russian).

Gasnikov A. V., Lagunovskaya A. A., Usmanova I. N., Fedorenko F. A., Krymova E. A. Stochastic online optimization. Single-point and multi-point non-linear multi-armed bandits. Convex and strongly-convex case // Automation and Remote Control. — 2017. — Vol. 78, No. 2. — P. 224–234.

Hazan E., Kale S. Beyond the regret minimization barrier: Optimal algorithms for stochastic strongly-convex optimization // JMLR. — 2014. — Vol. 15. — P. 2489–2512.

Hazan E. Introduction to online convex optimization // Foundations and Trends in Optimization. — 2015. — Vol. 2, No. 3–4. — P. 157–325.

Hao Yu, Neely M. J. , Xiaohan Wei. Online Convex Optimization with Stochastic Constraints // 2017. — https://arxiv.org/pdf/1708.03741.pdf

Jenatton R., Huang J., Archambeau C. Adaptive Algorithms for Online Convex Optimization with Long-term Constraints. — 2015. — https://arxiv.org/abs/1512.07422

Kalai A., Vempala S. Efficient algorithms for online decision problems // Journal of Computer and System Sciences. — 2005. — Vol. 71. — P. 291–307.

Lugosi G., Cesa-Bianchi N. Prediction, learning and games. — New York: Cambridge University Press, 2006. — 403 p.

Nemirovski A. Efficient methods for large-scale convex optimization problems // Ekonomika i Matematicheskie Metody. — 1979. — Vol. 15, No. 1 (in Russian).

Nemirovsky A., Yudin D. Problem Complexity and Method Efficiency in Optimization. — New York: J. Wiley & Sons, 1983. — 404 p.

Nesterov Yu. Introductory Lectures on Convex Optimization: A Basic Course. — Kluwer Academic Publishers, Massachusetts, 2004. — 236 p.

Polyak B. A general method of solving extremum problems // Soviet Mathematics Doklady. — 1967. — Vol. 8, No. 3. — P. 593–597 (in Russian).

Shor N. Z. Generalized gradient descent with application to block programming // Kibernetika. — 1967. — Vol. 3, No. 3. — P. 53–55 (in Russian).

Stonyakin F. S., Alkousa M. S., Stepanov A. N., Barinov M. A. Adaptive mirror descent algorithms in convex programming problems with Lipschitz constraints // Trudy Instituta Matematiki i Mekhaniki URO RAN. — 2018. — Vol. 24, No. 2. — P. 266–279.

Shpirko S., Nesterov Yu. Primal-dual subgradient methods for huge-scale linear conic problem // SIAM Journal on Optimization. — 2014. — Vol. 24, No. 3. — P. 1444–1457.

Titov A. A., Stonyakin F. S., Gasnikov A. V., Alkousa M. S. Mirror Descent and Constrained Online Optimization Problems // Optimization and Applications. 9th International Conference OPTIMA-2018 (Petrovac, Montenegro, October 1–5, 2018). Revised Selected Papers. Communications in Computer and Information Science. — 2019. — Vol. 974. — P. 64–78.

Zinkevich M. Online Convex Programming and Generalized Infinitesimal Gradient Ascent // Proceedings of International Conference on Machine Learning (ICML). — 2003.