КОМПЬЮТЕРНЫЕ ИССЛЕДОВАНИЯ
И МОДЕЛИРОВАНИЕ 2022 Т. 14 № 2 С. 445–472
DOI: 10.20537/2076-7633-2022-14-2-445-472

УДК: 519.8

Адаптивные методы первого порядка для относительно сильно выпуклых задач оптимизации

О. С. Савчук[1,a], А. А. Титов[2,3,b], Ф. С. Стонякин[1,2,c], М. С. Алкуса[2,3,d]

[1]Крымский федеральный университет им. В. И. Вернадского,
Россия, 295007, Республика Крым, г. Симферополь, проспект академика Вернадского, 4
[2]Московский физико-технический институт,
Россия, 141701, Московская обл., г. Долгопрудный, Институтский пер., 9
[3]Национальный исследовательский университет «Высшая школа экономики»,
Россия, 101000, г. Москва, ул. Мясницкая, д. 20

E-mail: [a] oleg.savchuk19@mail.ru, [b] a.a.titov@phystech.edu, [c] fedyor@mail.ru,
[d] mohammad.alkousa@phystech.edu

Получено 12.02.2022.
Принято к публикации 13.02.2022.

Настоящая статья посвящена некоторым адаптивным методам первого порядка для оптимизационных задач с относительно сильно выпуклыми функционалами. Недавно возникшее в оптимизации понятие относительной сильной выпуклости существенно расширяет класс выпуклых задач посредством замены в определении евклидовой нормы расстоянием в более общем смысле (точнее — расхождением или дивергенцией Брегмана). Важная особенность рассматриваемых в настоящей работе классов задач — обобщение стандартных требований к уровню гладкости целевых функционалов. Точнее говоря, рассматриваются относительно гладкие и относительно липшицевые целевые функционалы. Это может позволить применять рассматриваемую методику для решения многих прикладных задач, среди которых можно выделить задачу о нахождении общей точки системы эллипсоидов, а также задачу бинарной классификации с помощью метода опорных векторов. Если целевой функционал минимизационной задачи выпуклый, то условие относительной сильной выпуклости можно получить посредством регуляризации. В предлагаемой работе впервые предложены адаптивные методы градиентного типа для задач оптимизации с относительно сильно выпуклыми и относительно липшицевыми функционалами. Далее, в статье предложены универсальные методы для относительно сильно выпуклых задач оптимизации. Указанная методика основана на введении искусственной неточности в оптимизационную модель. Это позволило обосновать применимость предложенных методов на классе относительно гладких, так и на классе относительно липшицевых функционалов. При этом показано, как можно реализовать одновременно адаптивную настройку на значения параметров, соответствующих как гладкости задачи, так и введенной в оптимизационную модель искусственной неточности. Более того, показана оптимальность оценок сложности с точностью до умножения на константу для рассмотренных в работе универсальных методов градиентного типа для обоих классов относительно сильно выпуклых задач. Также в статье для задач выпуклого программирования с относительно липшицевыми функционалами обоснована возможность использования специальной схемы рестартов алгоритма зеркального спуска и доказана оптимальная оценка сложности такого алгоритма. Также приводятся результаты некоторых вычислительных экспериментов для сравнения работы предложенных в статье методов и анализируется целесообразность их применения.

Ключевые слова: адаптивный метод, относительно сильно выпуклый функционал, относительно гладкий функционал, относительно липшицев функционал, оптимальный метод, зеркальный спуск

Работа выполнена при поддержке Российского научного фонда, проект 21-71-30005.

COMPUTER RESEARCH AND MODELING
2022 VOL. 14 NO. 2 P. 445–472
DOI: 10.20537/2076-7633-2022-14-2-445-472

MATHEMATICAL MODELING AND NUMERICAL SIMULATION

UDC: 519.8

Adaptive first-order methods for relatively strongly convex optimization problems

O. S. Savchuk[1,a], A. A. Titov[2,3,b], F. S. Stonyakin[1,2,c], M. S. Alkousa[2,3,d]

[1]V. I. Vernadsky Crimean Federal University,
4 Academician Vernadsky Avenue, Simferopol, Republic of Crimea, 295007, Russia
[2]Moscow Institute of Physics and Technology,
9 Institutskiy per., Dolgoprudny, Moscow region, 141701, Russia
[3]HSE University,
20 Myasnitskaya st., Moscow, 101000, Russia

E-mail: [a] oleg.savchuk19@mail.ru, [b] a.a.titov@phystech.edu, [c] fedyor@mail.ru,
[d] mohammad.alkousa@phystech.edu

Received 12.02.2022.
Accepted for publication 13.02.2022.

The article is devoted to first-order adaptive methods for optimization problems with relatively strongly convex functionals. The concept of relatively strong convexity significantly extends the classical concept of convexity by replacing the Euclidean norm in the definition by the distance in a more general sense (more precisely, by Bregman's divergence). An important feature of the considered classes of problems is the reduced requirements concerting the level of smoothness of objective functionals. More precisely, we consider relatively smooth and relatively Lipschitz-continuous objective functionals, which allows us to apply the proposed techniques for solving many applied problems, such as the intersection of the ellipsoids problem (IEP), the Support Vector Machine (SVM) for a binary classification problem, etc. If the objective functional is convex, the condition of relatively strong convexity can be satisfied using the problem regularization. In this work, we propose adaptive gradient-type methods for optimization problems with relatively strongly convex and relatively Lipschitz-continuous functionals for the first time. Further, we propose universal methods for relatively strongly convex optimization problems. This technique is based on introducing an artificial inaccuracy into the optimization model, so the proposed methods can be applied both to the case of relatively smooth and relatively Lipschitz-continuous functionals. Additionally, we demonstrate the optimality of the proposed universal gradient-type methods up to the multiplication by a constant for both classes of relatively strongly convex problems. Also, we show how to apply the technique of restarts of the mirror descent algorithm to solve relatively Lipschitz-continuous optimization problems. Moreover, we prove the optimal estimate of the rate of convergence of such a technique. Also, we present the results of numerical experiments to compare the performance of the proposed methods.

Keywords: adaptive method, relatively strongly convex functional, relatively smooth functional, relatively Lipschitz-continuous functional, optimal method, mirror descent

Citation: *Computer Research and Modeling*, 2022, vol. 14, no. 2, pp. 445–472.
This work was supported by Russian Science Foundation (project No. 21-71-30005).

1. Introduction

Gradient-type numerical methods are often used for a wide variety of convex optimization problems in high-dimensional spaces. It can be explained by the low memory consumption at iterations, as well as the possibility of substantiating acceptable convergence rate estimates that do not contain (unlike, for example, the cutting hyperplane methods) space dimension parameters. However, some assumptions about the functional properties of such problems (smoothness, Lipschitz property, strong convexity, etc) are essential. For example, several years ago, there was introduced a class of relatively smooth convex optimization problems for which the convergence estimates for non-accelerated gradient-type methods are optimal up to the multiplication by a constant, which does not depend on the dimension and parameters of the method (see [Bauschke, Bolte, Teboulle, 2017; Dragomir et al., 2021; Dragomir, 2021; Lu, Freund, Nesterov, 2018] and their references). The concept of relatively strong convexity of a function is introduced in [Lu, Freund, Nesterov, 2018]. So, the class of convex optimization problems was extended for which the linear rate of convergence of gradient-type methods takes place (convergence with the rate of the geometric progression). Moreover, the corresponding estimate does not contain parameters of the dimension of the problem. In this paper, we develop this approach and introduce some algorithms for relatively strongly convex optimization problems. Recall that the relatively strong convexity [Lu, Freund, Nesterov, 2018] of a function f generalizes the concept of ordinary strong convexity by replacing in the following inequality (Q is the domain of f)

$$f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2}\|x - y\|_2^2 \leqslant f(y) \quad \forall x, y \in Q, \tag{1}$$

the term $\frac{\mu}{2}\|x - y\|_2^2$ by the Bregman divergence (see (2)), which is generated by some convex (not necessarily strongly convex) prox function

$$f(x) + \langle \nabla f(x), y - x \rangle + \mu V(y, x) \leqslant f(y) \quad \forall x, y \in Q. \tag{2}$$

The article consists of the introduction, conclusion, and six main sections. In Section 2, we give some auxiliary definitions and notations. Section 3 is devoted to the proposed adaptive algorithms for relatively strongly convex and relatively Lipschitz-continuous optimization problems. Note that the approach is somewhat reminiscent of universal gradient methods associated with the idea of introducing an artificial inaccuracy into the optimization model in order to apply the method to both smooth and nonsmooth cases [Nesterov, 2015]. Nevertheless, there is also a significant difference — no need to use the value of the objective function in the stopping criteria from iterations. In Section 4, we propose universal algorithms for relatively strongly convex minimization problems, applicable to both relatively smooth and relatively Lipschitz-continuous problems. It is also shown that these methods lead to optimal complexity estimates for both classes of problems and adaptively select the smoothness class itself. In Section 5, we introduce a corresponding variation of the proposed method with an analogue of the (δ, L, μ)-oracle [Devolder, Glineur, Nesterov, 2013], which makes it possible to apply the proposed approach in the case where we know only inaccurate information about the objective function. Moreover, it is shown that such inaccuracies do not accumulate in convergence rate estimates. In Section 6, we consider convex programming problems with relatively strongly convex and relatively Lipschitz-continuous objective and functional constraints. For such problems, we propose the restarted technique of the mirror descent method with switchings between productive and nonproductive steps. We show that such a structure guarantees an optimal estimate of the convergence rate for the corresponding class of problems. Section 7 is devoted to numerical experiments for the problem of minimizing a relatively strongly convex functional in order to compare the proposed approaches and illustrate the effectiveness of the obtained theoretical estimates with adaptively selected parameters.

Summing up, the contribution of this paper is as follows.

- We propose adaptive methods for relatively strongly convex and relatively Lipschitz-continuous problems and investigate their theoretical estimates of the convergence rate.

- We also introduce adaptive methods for relatively Lipschitz-continuous and relatively strongly convex problems with functional constraints.

- We propose universal methods for relatively strongly convex problems for the case of both relatively smooth and relatively Lipschitz-continuous objectives.

- We introduce the definition of an analogue of the (δ, L, μ)-oracle and propose the corresponding numerical algorithms for solving problems with such a class of smoothness.

2. Basic definitions and notations

In this paper, we consider the following optimization problem:

$$\min_{x \in Q} f(x) \tag{3}$$

for a convex objective f and a closed convex set $Q \subset \mathbb{R}^n$. Let us give some basic definitions and notations concerning Bregman divergence and the prox structure, which will be used throughout the paper. Let $(E, \|\cdot\|)$ be some normed finite-dimensional vector space and E^* be its conjugate space with the norm

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

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

Let $d\colon Q \to \mathbb{R}$ be a non-negative distance generating function (d.g.f) which is continuously differentiable and convex. Assume that $\min\limits_{x \in Q} d(x) = d(0)$ and suppose that there exists a constant $\Theta_0 > 0$, such that $d(x_*) \leqslant \Theta_0^2$, where x_* is a solution of the problem (3) (supposing that the problem (3) is solvable). Note that, if there is a set of optimal points $X_* \subset Q$, we assume that $\min\limits_{x_* \in X_*} d(x_*) \leqslant \Theta_0^2$.

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

$$V(y, x) = d(y) - d(x) - \langle \nabla d(x), y - x\rangle. \tag{4}$$

This article is devoted to adaptive methods for relatively strongly convex problems with the following condition of the relatively Lipschitz-continuity (also known as relative continuity or M-relatively Lipschitz-continuity), proposed recently in [Lu, 2019; Nesterov, 2019b]

$$\langle \nabla f(x), y - x\rangle + M\sqrt{2V(y, x)} \geqslant 0 \quad \forall x, y \in Q. \tag{5}$$

Let us note that for each $\varepsilon > 0$, $L = \frac{M^2}{\varepsilon}$ and $\delta = \frac{\varepsilon}{2}$ the inequality

$$\langle \nabla f(x), y - x\rangle + LV(y, x) + \delta \geqslant 0 \quad \forall x, y \in Q \tag{6}$$

follows from (5).

Algorithm 1. Adaptive Algorithm for relatively Lipschitz-continuous optimization problems

Require: $\varepsilon > 0$, x_0, $L_0 > 0$, R s.t. $V(x_*, x_0) \leqslant R^2$.

1: Set $k = k + 1$, $L_{k+1} = \frac{L_k}{2}$.

2: Find

$$x_{k+1} = \arg\min_{x \in Q}\{\langle \nabla f(x_k), x\rangle + L_{k+1}V(x, x_k)\}. \tag{7}$$

3: **if**

$$0 \leqslant \langle \nabla f(x_k), x_{k+1} - x_k\rangle + L_{k+1}V(x_{k+1}, x_k) + \frac{\varepsilon}{2}, \tag{8}$$

then go to the next iteration (item 1).

4: **else**

set $L_{k+1} = 2 \cdot L_{k+1}$ and go to item 2.

5: **end if**

3. Adaptive algorithms for Relatively Lipschitz-continuous convex optimization problems

In this section, we present two algorithms that implement the adaptive adjustment of the parameter L from (6). In addition, Algorithm 2 has an adaptive setting for both L and the artificial inaccuracy parameter δ, which can potentially improve the quality of the output solution.

Theorem 1. *Let f be a relatively strongly convex functional* (2) *and $L_i \geqslant \mu$ for each* $0 < i \leqslant N$. *Then after the stopping of Algorithm* 1, *the following inequality holds*:

$$\min_{i=\overline{0, N-1}} f(x_i) - f(x_*) \leqslant \min\left\{\max\left\{0;\ L_N\prod_{i=1}^{N}\left(1-\frac{\mu}{L_i}\right)\right\},\ \frac{1}{\sum\limits_{i=1}^{N}\frac{1}{L_i}}\right\}V(x_*, x_0) + \frac{\varepsilon}{2}. \tag{9}$$

Moreover, if f is M-relatively Lipschitz-continuous, then the estimate $\min\limits_{i=\overline{0, N-1}} f(x_i) - f(x_*) \leqslant \varepsilon$ holds after no more than

$$N = O\left(\frac{M^2}{\mu\varepsilon}\log\frac{1}{\varepsilon}\right) \tag{10}$$

iterations of Algorithm 1.

The proof of Theorem 1 is given in Appendix A.

The following theorem describes the properties of Algorithm 2.

Theorem 2. *Let f be a relatively strongly convex functional* (2) *and $L_i \geqslant \mu$ for each* $0 < i \leqslant N$. *Then after the stopping of Algorithm* 2, *the following inequality holds*:

$$\min_{i=\overline{0, N-1}} f(x_i) - f(x_*) \leqslant L_N\prod_{i=1}^{N}\left(1-\frac{\mu}{L_i}\right)V(x_*, x_0) + \frac{1}{\widehat{S}_N}\sum_{i=1}^{N}\frac{\delta_i q_i}{L_i}, \tag{13}$$

where $q_i = \prod\limits_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)$ for each $i < N$, $q_N = 1$ and $\widehat{S}_N := \sum\limits_{i=1}^{N}\frac{q_i}{L_i}$. Moreover, if f is M-relatively Lipschitz-continuous, then the estimate $\min\limits_{i=\overline{0, N-1}} f(x_i) - f(x_*) \leqslant \varepsilon$ holds after no more then

$$N = O\left(\frac{M^2}{\mu\varepsilon}\log\frac{1}{\varepsilon}\right) \tag{14}$$

iterations of Algorithm 2.

Algorithm 2. Adaptation to inexactness for relatively Lipschitz-continuous optimization problems

Require: $\varepsilon > 0$, x_0, $L_0 > 0$, $\delta_0 > 0$, R s.t. $V(x_*, x_0) \leqslant R^2$.

1: Set $k = k + 1$, $L_{k+1} = \frac{L_k}{2}$, $\delta_{k+1} = \frac{\delta_k}{2}$.

2: Find

$$x_{k+1} = \arg\min_{x \in Q}\{\langle \nabla f(x_k), x\rangle + L_{k+1} V(x, x_k)\}. \tag{11}$$

3: **if**

$$0 \leqslant \langle \nabla f(x_k), x_{k+1} - x_k\rangle + L_{k+1} V(x_{k+1}, x_k) + \delta_{k+1}, \tag{12}$$

then go to the next iteration (item 1).

4: **else**

set $L_{k+1} = 2 \cdot L_{k+1}$, $\delta_{k+1} = 2 \cdot \delta_{k+1}$ and go to item 2.

5: **end if**

The proof of Theorem 2 is given in Appendix B.

Remark 1. For M-relatively Lipschitz-continuous f and $L_{k+1} \geqslant L = \frac{M^2}{\varepsilon}$, $\delta_{k+1} \geqslant \delta = \frac{\varepsilon}{2}$ we have (12). So, for $C = \max\left\{\frac{2L}{L_0}; \frac{2\delta}{\delta_0}\right\}$, $L_{k+1} \leqslant CL$ and $\delta_{k+1} \leqslant C\delta = \frac{C\varepsilon}{2}$ $\forall k \geqslant 0$. Thus, $\min\limits_{i=0,N-1} f(x_i) - f(x_*) \leqslant \varepsilon$ after (14) iterations of Algorithm 2. This fact, in essence, substantiates the optimality of the proposed method for the class of optimization problems with relatively strongly convex and relatively Lipschitz-continuous functions.

4. Universal gradient-type methods for relatively strongly convex functions

Let us note that in the previous section we justified the applicability and optimality of Algorithms 1 and 2 only for the class of relatively Lipschitz-continuous problems. Now we consider universal variants of these methods, which are applicable to both relatively Lipschitz-continuous and relatively smooth problems.

Algorithm 3. Universal method for relatively smooth and Lipschitz-continuous optimization problems with adaptation to inexactness

Require: $\varepsilon > 0$, x_0, $L_0 > 0$, $\delta_0 > 0$, R s.t. $V(x_*, x_0) \leqslant R^2$.

1: Set $k = k + 1$, $L_{k+1} = \frac{L_k}{2}$, $\delta_{k+1} = \frac{\delta_k}{2}$.

2: Find

$$x_{k+1} = \arg\min_{x \in Q}\{\langle \nabla f(x_k), x\rangle + L_{k+1} V(x, x_k)\}. \tag{15}$$

3: **If**

$$f(x_{k+1}) \leqslant f(x_k) + \langle \nabla f(x_k), x_{k+1} - x_k\rangle + L_{k+1} V(x_{k+1}, x_k) + \delta_{k+1}, \tag{16}$$

then go to the next iteration (item 1).

4: **else**

set $L_{k+1} = 2 \cdot L_{k+1}$, $\delta_{k+1} = 2 \cdot \delta_{k+1}$ and go to item 2.

5: **end if**

Theorem 3. *Let f be a relatively μ-strongly convex functional* (2) *and $L_i \geqslant \mu$ for each $0 < i \leqslant N$. Then after the stopping of Algorithm* 3*, the following inequality holds*:

$$\min_{i=\overline{1,N}} f(x_i) - f(x_*) \leqslant L_N \prod_{i=1}^{N}\left(1 - \frac{\mu}{L_i}\right) V(x_*, x_0) + \frac{1}{\widehat{S}_N} \sum_{i=1}^{N} \frac{\delta_i q_i}{L_i}, \tag{17}$$

where $q_i = \prod_{n=i+1}^{N}\left(1 - \frac{\mu}{L_n}\right)$ *for each* $i < N$, $q_N = 1$ *and* $\widehat{S}_N := \sum_{i=1}^{N} \frac{q_i}{L_i}$. *Moreover, if* f *is* M*-relatively Lipschitz-continuous, then the estimate* $\min_{i=\overline{1,N}} f(x_i) - f(x_*) \leqslant \varepsilon$ *holds after no more than*

$$N = O\left(\frac{M^2}{\mu\varepsilon}\log\frac{1}{\varepsilon}\right)$$

iterations of Algorithm 3.

The proof of Theorem 3 is given in Appendix C.

The optimality of Algorithm 3 for the class of relatively strongly convex and M-relatively Lipschitz-continuous problems can be proved similarly to Remark 1. The optimal rate of convergence $O(\log\varepsilon^{-1})$ for the class of relatively strongly convex and L-relatively smooth problems takes place for Algorithm 3, as L does not depend on ε.

Let us now formulate a variant of the universal method for relatively Lipschitz-continuous and relatively smooth problems, which makes it possible to prove the guaranteed preservation of the optimal complexity estimates. This method is listed as Algorithm 4 below.

Algorithm 4. Universal method for relatively smooth and Lipschitz-continuous optimization problems

Require: $\varepsilon > 0$, x_0, $L_0 > 0$, R s.t. $V(x_*, x_0) \leqslant R^2$.

1: Set $k = k + 1$, $L_{k+1} = \frac{L_k}{2}$.

2: Find

$$x_{k+1} = \arg\min_{x\in Q}\{\langle\nabla f(x_k), x\rangle + L_{k+1}V(x, x_k)\}. \tag{18}$$

3: **If**

$$f(x_{k+1}) \leqslant f(x_k) + \langle\nabla f(x_k), x_{k+1} - x_k\rangle + L_{k+1}V(x_{k+1}, x_k) + \frac{3\varepsilon}{4}, \tag{19}$$

then go to the next iteration (item 1).

4: **else**

set $L_{k+1} = 2 \cdot L_{k+1}$ and go to item 2.

5: **end if**

Theorem 4. *Let* f *be a relatively strongly convex functional* (2) *and* $L_i \geqslant \mu$ *for each* $0 < i \leqslant N$. *Then after the stopping of Algorithm* 4, *the following inequality holds*:

$$\min_{i=\overline{1,N}} f(x_i) - f(x_*) \leqslant \min\left\{\max\left\{0; L_N\prod_{i=1}^{N}\left(1 - \frac{\mu}{L_i}\right)\right\}, \frac{1}{\sum_{i=1}^{N}\frac{1}{L_i}}\right\}V(x_*, x_0) + \frac{3\varepsilon}{4}. \tag{20}$$

Moreover, if f *is* M*-relatively Lipschitz-continuous, then the estimate* $\min_{i=\overline{1,N}} f(x_i) - f(x_*) \leqslant \varepsilon$ *holds after no more than*

$$N = O\left(\frac{M^2}{\mu\varepsilon}\log\frac{1}{\varepsilon}\right)$$

iterations of Algorithm 4.

The proof of Theorem 4 is given in Appendix D.

REMARK 2. Note that for relatively smooth f Algorithm 4 has the linear rate of convergence. Indeed, for $L_{k+1} \geqslant L$, (19) holds. So, $L_{k+1} \leqslant 2L$, $\forall k \geqslant 0$. Therefore, the following inequalities hold:

$$L_N \prod_{i=1}^{N} \left(1 - \frac{\mu}{L_i}\right) \leqslant 2L\left(1 - \frac{\mu}{2L}\right)^N,$$

and

$$\max\left\{0;\ L_N \prod_{i=1}^{N} \left(1 - \frac{\mu}{L_i}\right)\right\} V(x_*, x_0) \leqslant 2LV(x_*, x_0)\left(1 - \frac{\mu}{2L}\right)^N \leqslant \frac{\varepsilon}{4}.$$

Thus, the estimate

$$\min_{i=\overline{1,N}} f(x_i) - f(x_*) \leqslant \varepsilon$$

holds after no more than

$$N = O\left(\log \frac{1}{\varepsilon}\right) \tag{21}$$

iterations of Algorithm 4.

5. Variant of inexact oracle for relatively strongly convex problems

In the previous section, we presented universal methods applicable to both relatively Lipschitz-continuous and relatively smooth problems. However, the disadvantage of such approaches is the need to know the value of the objective function at the current point. In this section, we consider a generalization of this technique to problems which allow using an inexact oracle. Thus, we can solve optimization problems without complete knowledge about the objective function or even its gradient.

Definition 1. Let f be a functional defined on a convex set Q in a normed finite-dimensional vector space $\mathbb{E}$. We will say that f is equipped with a first-order (δ, L, V, μ)-oracle if for any $x \in Q$ we can compute a pair $(f_\delta(x), g_\delta(x)) \in \mathbb{R} \times \mathbb{E}^*$ such that

$$\mu V(y, x) \leqslant f(y) - (f_\delta(x) + \langle g_\delta(x), y - x\rangle) \leqslant LV(y, x) + \delta, \tag{22}$$

for all $x \in Q$, where $\delta \geqslant 0$ and $L \geqslant \mu > 0$.

In particular, if $f_\delta(x) = f(x)$ and $g_\delta(x) = \nabla f(x)$, then (22) is the condition of relatively strong convexity of f (2). Let us note that $y = x$ means that for any $x \in Q$ the following statement holds $f_\delta(x) \in$ $\in [f(x) - \delta;\ f(x)]$.

For Algorithm 5, we have the following result.

Theorem 5. *Let f be equipped with a first-order (δ, L, V, μ)-oracle (22) and $L_i \geqslant \mu > 0$ for each $0 < i \leqslant N$. Then, after the stopping of Algorithm 5, the following inequality holds*:

$$\min_{i=\overline{1,N}} f(x_i) - f(x_*) \leqslant L_N \prod_{i=1}^{N} \left(1 - \frac{\mu}{L_i}\right) V(x_*, x_0) + \frac{\delta}{\widehat{S}_N} \sum_{i=1}^{N} \frac{q_i}{L_i} + \frac{1}{\widehat{S}_N} \sum_{i=1}^{N} \frac{\delta_i q_i}{L_i}, \tag{25}$$

where $q_i = \prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)$ for each $i < N$, $q_N = 1$ and $\widehat{S}_N := \sum_{i=1}^{N} \frac{q_i}{L_i}$.

Moreover, if $\delta = 0$ and f is M-relatively Lipschitz-continuous, then the estimate $\min_{i=\overline{1,N}} f(x_i) -$ $- f(x_*) \leqslant \varepsilon$ holds after no more than

$$N = O\left(\frac{M^2}{\mu\varepsilon} \log \frac{1}{\varepsilon}\right)$$

iterations of Algorithm 5.

The proof of Theorem 5 is given in Appendix E.

Algorithm 5. Universal method for (δ, L, V, μ)-oracle for relatively strongly convex optimization problems with adaptation to inexactness

Require: $\varepsilon > 0$, x_0, $L_0 > 0$, $\delta_0 > 0$, R s.t. $V(x_*, x_0) \leqslant R^2$.

1: Set $k = k + 1$, $L_{k+1} = \frac{L_k}{2}$, $\delta_{k+1} = \frac{\delta_k}{2}$.

2: Find

$$x_{k+1} = \arg\min_{x \in Q} \{\langle g_\delta(x_k), x\rangle + L_{k+1} V(x, x_k)\}. \tag{23}$$

3: **If**

$$f_\delta(x_{k+1}) \leqslant f_\delta(x_k) + \langle g_\delta(x_k), x_{k+1} - x_k\rangle + L_{k+1} V(x_{k+1}, x_k) + \delta_{k+1}, \tag{24}$$

then go to the next iteration (item 1).

4: **else**

$$\text{set } L_{k+1} = 2 \cdot L_{k+1},\ \delta_{k+1} = 2 \cdot \delta_{k+1} \text{ and go to item 2.}$$

5: **end if**

6. Restarted procedures for relatively strongly convex programming problems

In this section, we consider the following optimization problem with a functional constraint:

$$\min_{x \in Q} f(x) \tag{26}$$

$$\text{s.t. } g(x) \leqslant 0, \tag{27}$$

under the assumption that f and g are relatively strongly convex and M_f, M_g are relatively Lipschitz-continuous functionals and the Slaiter's condition holds for (26)–(27).

In order to solve the problem (26)–(27) we apply the restarted technique to the mirror descent algorithm (see Algorithm 6, below). The proposed restarted version of Algorithm 6 is listed as Algorithm 7 below. A distinctive feature of this section is that we consider the optimization problem with the functional constraint and avoid a logarithmic factor in the estimation of the complexity.

Theorem 6. *Let f and g be relatively strongly convex functions, x_* be a solution of the problem* (26)–(27) *and $M = \max\{M_f, M_g\}$. Then Algorithm* 7 *provides an ε-solution*

$$f(x_p) - f(x_*) \leqslant \varepsilon, \quad g(x_p) \leqslant \varepsilon$$

of the problem (26)–(27). *Moreover, the total number of iterations of Algorithm* 6 *will not exceed $O\left(\frac{M^2}{\mu\varepsilon}\right)$.*

The proof of Theorem 6 is given in Appendix F.

7. Numerical experiments for relatively strongly convex functionals

In this section we consider some numerical experiments in order to solve the problem (3) with a relatively μ-strongly convex objective function. We consider the following objective:

$$f(x) := \frac{1}{4}\|Bx\|_2^4 + \frac{1}{4}\|Ax - b\|_4^4 + \frac{1}{2}\|Cx - \widehat{b}\|_2^2, \tag{28}$$

where $B, A, C \in \mathbb{R}^{n \times n}$ and $b, \widehat{b} \in \mathbb{R}^n$.

Let $\sigma_B > 0$ and $\sigma_C > 0$ denote the smallest singular values of B and C, respectively. The objective function (28) is M-smooth and μ-strongly convex with respect to the following prox function:

$$d(x) := \frac{1}{4}\|x\|_2^4 + \frac{1}{2}\|x\|_2^2 \tag{29}$$

Algorithm 6. Mirror descent for relatively Lipschitz-continuous functions (Algorithm 3, [Titov et al., 2020])

Require: $\varepsilon > 0$, $M_f > 0$, $M_g > 0$, $\Theta_0 > 0$ s.t. $V(x^*, x_0) \leqslant \Theta_0^2$.

$I = \varnothing$, $J = \varnothing$, $N = 0$.
repeat
  if $g(x_N) \leqslant \varepsilon$ **then**
    $h^f = \frac{\varepsilon}{M_f^2}$,
    $x_{N+1} = Mirr_{h^f}(x_N, \nabla f)$, «productive step»
    $I = I \cup \{N\}$.
  else
    $h^g = \frac{\varepsilon}{M_g^2}$,
    $x_{N+1} = Mirr_{h^g}(x_N, \nabla g)$, «nonproductive step»
    $J = J \cup \{N\}$.
  end if
  $N = N + 1$.
until $\frac{2\Theta_0^2}{\varepsilon^2} \leqslant \frac{|I|}{M_f^2} + \frac{|J|}{M_g^2}$.

Ensure: $\widehat{x} = \frac{1}{|I|} \sum\limits_{k \in I} x_k$.

Algorithm 7. Restarted procedure for Algorithm 6

Require: $\varepsilon > 0$, $M_f > 0$, $M_g > 0$, $\mu > 0$, $\Omega > 0$ s.t. $d(\cdot) \leqslant \frac{\Omega}{2}$, x_0, R_0 s.t. $V(x_0, x^*) \leqslant R_0^2$.

$d_0(x) = d\left(\frac{x - x_0}{R_0}\right)$, $p = 1$.
repeat
  Set $R_p^2 = R_0^2 \cdot 2^{-p}$, $\varepsilon_p = \mu R_p^2$.
  Set x_p as the output of Algorithm 6, with accuracy ε_p, prox-function $d_{p-1}(\cdot)$ and $\frac{\Omega}{2}$ as Θ_0^2.
  $d_p(x) = d\left(\frac{x - x_p}{R_p}\right)$.
  $p = p + 1$.
until $p > \log_2 \frac{\mu R_0^2}{\varepsilon}$.

Ensure: x_p.

where $M = 3\|B\|^4 + 3\|A\|^4 + 6\|A\|^3\|b\|_2 + 3\|A\|^2\|b\|_2^2 + \|C\|^2$ and $\mu = \min\left\{\frac{\sigma_B^4}{3}, \sigma_C^2\right\}$ [Lu, Freund, Nesterov, 2018] (the norm of the matrices is taken with respect to the ℓ_2 (spectral) norm).

Each iteration of Algorithms 1, 2, 3, and 4 requires the capability to solve the subproblem (7) (which is the same as (11), (15), and (18)). This subproblem is equivalent to the following linearized problem:

$$x_{k+1} = \arg\min_{x \in Q}\{\langle c_k, x\rangle + d(x)\}, \tag{30}$$

where $c_k = \frac{1}{L_{k+1}} \nabla f(x_k) - \nabla d(x_k)$ and $d(x)$ is given in (29). The solution of this subproblem can be found explicitly

$$x_{k+1} = -\theta_k c_k, \tag{31}$$

for some $\theta_k \geqslant 0$, where θ_k is a positive real root of the following cubic equation:

$$1 - \theta - \|c_k\|_2^2 \theta^3 = 0. \tag{32}$$

We run Algorithms 1, 2, 3, and 4 with the prox function (29) and the starting point $\left(\frac{1}{\sqrt{n}}, \ldots, \frac{1}{\sqrt{n}}\right) \in \mathbb{R}^n$. The matrices A, B, C and vectors b, $\widehat{b}$ are chosen randomly from the normal (Gaussian) distribution. We also take $L_0 = \frac{\|\nabla f(1,0,\ldots,0) - \nabla f(0,1,0,\ldots,0)\|_2}{\sqrt{2}}$, $\delta_0 = 0{,}5$, $\varepsilon = 0{,}01$, and Q as the unit ball.

The results of the work of Algorithms 1, 2, 3, and 4 are presented in Table 1, for $n = 1000$ and different numbers of iterations k. These results demonstrate the running time in seconds as a function of the number of iterations, and the quality of the solution «Estimate», which is the right-hand side of inequality (20) for Algorithm 4 and for Algorithm 1, but with $\frac{\varepsilon}{2}$ instead of $\frac{3\varepsilon}{4}$. For the estimate of Algorithms 2 and 3 see the right-hand side of inequality (26) in [Stonyakin et al., 2021b]

Table 1. The results of Algorithms 1, 2, 3 and 4 for the relatively strongly convex function (28), with $n = 1000$

k	Algorithm 1		Algorithm 2	
	Time (s)	Estimate	Time (s)	Estimate
10 000	95,355	1 303 048,197 941	107,331	7,646 420
15 000	143,265	1 302 121,012 456	149,001	4,380 504
20 000	191,214	1 301 025,001 257	192,637	3,071 235
25 000	238,397	1 291 571,125 478	239,461	2,365 555
30 000	286,537	1 281 257,332 561	287,485	1,924 260
35 000	334,011	1 280 012,001 253	335,497	1,622 212
40 000	382,275	1 276 521,884 556	384,625	1,402 485
45 000	420,254	1 273 260,112 543	431,755	1,235 462
50 000	482,460	1 267 521,001 156	479,466	1,104 210
55 000	539,106	1 265 231,012 546	526,251	0,998 350
60 000	588,012	1 262 543,112 473	573,579	0,911 163
65 000	637,003	1 252 145,001 254	670,067	0,838 109
70 000	687,168	1 250 015,953 264	708,150	0,776 008
80 000	785,822	1 240 008,875 231	814,882	0,676 100
90 000	882,865	1 232 457,210 365	922,103	0,599 241
100 000	1032,974	1 222 140,120 352	1000,121	0,538 280
k	Algorithm 3		Algorithm 4	
	Time (s)	Estimate	Time (s)	Estimate
10 000	139,996	5,232 175	130,952	1,760 732
15 000	210,958	3,464 171	196,467	1,168 329
20 000	281,926	2,590 491	262,126	0,875 128
25 000	349,203	2,069 540	327,275	0,700 226
30 000	420,262	1,723 586	393,057	0,583 960
35 000	462,928	1,477 132	459,710	0,501 132
40 000	523,621	1,292 649	527,592	0,439 108
45 000	589,625	1,149 374	590,131	0,390 942
50 000	654,889	1,034 887	654,781	0,352 438
55 000	720,028	0,941 303	720,073	0,320 961
60 000	785,231	0,863 377	786,581	0,294 758
65 000	850,083	0,797 482	850,766	0,272 585
70 000	916,504	0,741 019	917,232	0,253 598
80 000	1046,380	0,649 352	1047,675	0,222 758
90 000	1177,368	0,578 108	1178,440	0,198 785
100 000	1308,821	0,521 145	1310,018	0,179 617

From the results presented in Table 1, we can see that the Universal Algorithm 4, as compared to the other proposed Adaptive and Universal algorithms, gives the best quality of the solution, but it needs more time to achieve the solution. Also, we can see that the required time of work of the

Adaptive Algorithm 2, for $k = 100\,000$ iterations, is lower than the time of work of other algorithms. In addition, we note that the Adaptive Algorithm 1 gives the worst quality of the solution and it grows very slowly with the growth in the number of iterations.

8. Conclusions

In this article, for the first time, we have proposed adaptive methods for a class of relatively strongly convex and relatively Lipschitz-continuous problems. Moreover, some of these methods are universal in the sense that they give optimal complexity estimates for both relatively smooth and relatively Lipschitz-continuous problems. The use of adaptively selected parameters in convergence rate estimates allows using the developed algorithms without a priori knowledge of information about the smoothness properties of the problem (this information is important only for theoretical guarantees of the optimal convergence rate on a selected class, it is not important for the practical application). We have analyzed the results of the given numerical experiments and compared the effectiveness of the proposed algorithms with each other.

References

Antonakopoulos K., Mertikopoulos P. Adaptive first-order methods revisited: Convex optimization without Lipschitz requirements // Advances in Neural Information Processing Systems 34 pre-proceedings. Proceedings of the 2021 Conference / eds. M. Ranzato, A. Beygelzimer, K. Nguyen, P. S. Liang, J. W. Vaughan, Y. Dauphin. — The MIT Press, 2021.

Antonakopoulos K., Belmega E. V., Mertikopoulos P. An adaptive mirror-prox algorithm for variational inequalities with singular operators // Advances in Neural Information Processing Systems 32. Proceedings of the 2019 Conference / eds. H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett. — The MIT Press, 2019.

Bauschke H. H., Bolte J., Teboulle M. A descent lemma beyond Lipschitz gradient continuity: first-order methods revisited and applications // Mathematics of Operations Research. — 2017. — Vol. 42, No. 2. — P. 330–348.

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

Cohen M. B., Sidford A., Tian K. Relative Lipschitzness in extragradient methods and a direct recipe for acceleration // arXiv preprint. — 2020. — https://arxiv.org/pdf/2011.06572

Devolder O., Glineur F., Nesterov Yu. First-order methods with inexact oracle: the strongly convex case // LIDAM Discussion Papers CORE 2013016, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE). — 2013. — http://hdl.handle.net/2078.1/128723

Dragomir R. A., Taylor A. B., d'Aspremont A., Bolte J. Optimal complexity and certification of Bregman first-order methods // Mathematical Programming. — 2021. — P. 1–43. — https://doi.org/10.1007/s10107-021-01618-1

Dragomir R. A. Relatively-Smooth Optimization. — Doctoral dissertation. — UT1 Capitole, 2021. — https://hal.inria.fr/tel-03389344/document

Gasnikov A. V., Gorbunov E. A., Kovalev D. A., Mohammed A. A. M., Chernousova E. O. Substantiation of the hypothesis about optimal estimates of the rate of convergence of numerical methods of high-order convex optimization // Computer Research and Modeling. — 2018. — Vol. 10, No. 6. — P. 737–753.

Lu H., Freund R., Nesterov Yu. Relatively smooth convex optimization by first-order methods, and applications // SIAM Journal on Optimization. — 2018. — Vol. 28, No. 1. — P. 333–354.

Lu H. Relative continuity for non-Lipschitz nonsmooth convex optimization using stochastic (or deterministic) mirror descent // Informs Journal on Optimization. — 2019. — Vol. 1, No. 4. — P. 288–303.

Nesterov Yu. Gradient methods for minimizing composite functions // Mathematical Programming. — 2013. — Vol. 140, No. 1. — P. 125–161.

Nesterov Yu. Implementable tensor methods in unconstrained convex optimization // Mathematical Programming. — 2019. — Vol. 186. — P. 157–183.

Nesterov Yu. Relative Smoothness: New Paradigm in Convex Optimization // Conference report, EUSIPCO-2019, A Coruna, Spain. — 2019b. — Vol. 4.

Nesterov Yu. Inexact accelerated high-order proximal-point methods // Mathematical Programming. — 2021. — https://doi.org/10.1007/s10107-021-01727-x

Shwartz S. S., Singer Y., Srebro N., Cotter A. Pegasos: primal estimated sub-gradient solver for SVM // Mathematical Programming. — 2011. — Vol. 127. — P. 3–30.

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.

Stonyakin F., Tyurin A., Gasnikov A., Dvurechensky P., Agafonov A., Dvinskikh D., Alkousa M., Pasechnyuk D., Artamonov S., Piskunova V. Inexact relative smoothness and strong convexity for optimization and variational inequalities by inexact model // Optimization Methods and Software. — 2021. — https://doi.org/10.1080/10556788.2021.1924714

Stonyakin F., Titov A., Alkousa M., Savchuk O., Pasechnyuk D. Gradient-type adaptive methods for relatively Lipschitz convex optimization problems // arXiv preprint. — 2021b. — https://arxiv.org/pdf/2107.05765

Titov A. A., Stonyakin F. S., Alkousa M. S., Ablaev S. S., Gasnikov A. V. Analogues of switching subgradient schemes for relatively Lipschitz continuous convex programming problems // *Kochetov Y., Bykadorov I., Gruzdeva T.* (eds.) Mathematical optimization theory and operations research. — MOTOR 2020. Communications in Computer and Information Science. — Vol. 1275. — Springer, Cham. — https://doi.org/10.1007/978-3-030-58657-7_13

Titov A., Stonyakin F., Alkousa M., Gasnikov A. Algorithms for solving variational inequalities and saddle point problems with some generalizations of Lipschitz property for operators // *Strekalovsky A., Kochetov Y., Gruzdeva T., Orlov A.* (eds.) Mathematical optimization theory and operations research: recent trends. — MOTOR 2021. Communications in Computer and Information Science. — Vol. 1476. — Springer, Cham. — https://doi.org/10.1007/978-3-030-86433-0_6

Appendix A. The proof of Theorem 1

Due to (7) we have

$$\langle \nabla f(x),\, x_{k+1} - x\rangle \leqslant \langle \nabla f(x_k),\, x_{k+1} - x\rangle \leqslant L_{k+1} V(x,\, x_k) - L_{k+1} V(x,\, x_{k+1}) - L_{k+1} V(x_{k+1},\, x_k)$$

and

$$\begin{gathered} L_{k+1} V(x,\, x_k) - L_{k+1} V(x,\, x_{k+1}) \geqslant \langle \nabla f(x_k),\, x_{k+1} - x\rangle + L_{k+1} V(x_{k+1},\, x_k) = \\ = \langle \nabla f(x_k), x_{k+1} - x\rangle + \langle \nabla f(x_k), x_{k+1} - x_k\rangle + L_{k+1} V(x_{k+1}, x_k) - \langle \nabla f(x_k), x_{k+1} - x_k\rangle, \\ \langle \nabla f(x_k),\, x_{k+1} - x\rangle + \langle \nabla f(x_k),\, x_{k+1} - x_k\rangle + L_{k+1} V(x_{k+1},\, x_k) - \langle \nabla f(x_k), x_{k+1} - x_k\rangle \geqslant \\ \geqslant \langle \nabla f(x_k),\, x_{k+1} - x\rangle - \langle \nabla f(x_k),\, x_{k+1} - x_k\rangle - \frac{\varepsilon}{2} = \langle \nabla f(x_k),\, x_k - x\rangle - \frac{\varepsilon}{2}. \end{gathered}$$

Thus, the following inequality holds:

$$\langle \nabla f(x_k), x_k - x\rangle \leqslant L_{k+1}V(x, x_k) - L_{k+1}V(x, x_{k+1}) + \frac{\varepsilon}{2}. \tag{33}$$

Combining this inequality with

$$\langle \nabla f(x_k), x_k - x\rangle \geqslant f(x_k) - f(x) + \mu V(x, x_k),$$

we have from (2)

$$f(x_k) - f(x) + \mu V(x, x_k) \leqslant L_{k+1}V(x, x_k) - L_{k+1}V(x, x_{k+1}) + \frac{\varepsilon}{2}. \tag{34}$$

After simplifying and combining terms, we rewrite (34) in the following form:

$$L_{k+1}V(x, x_{k+1}) \leqslant (L_{k+1} - \mu)V(x, x_k) + f(x) - f(x_k) + \frac{\varepsilon}{2}, \quad \forall x \in Q. \tag{35}$$

Further, let us rewrite (35) for $x = x_*$ as

$$L_{k+1}V(x_*, x_{k+1}) \leqslant (L_{k+1} - \mu)V(x_*, x_k) + f(x_*) - f(x_k) + \frac{\varepsilon}{2}. \tag{36}$$

By dividing both sides of inequality (36) by L_{k+1}, we have

$$V(x_*, x_{k+1}) \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)V(x_*, x_k) + \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_k)\Big) + \frac{\varepsilon}{2L_{k+1}}. \tag{37}$$

Continuing inequality (37) recursively, we find that

$$\begin{gathered}
V(x_*, x_{k+1}) \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)V(x_*, x_k) + \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_k)\Big) + \frac{\varepsilon}{2L_{k+1}} \leqslant \\
\leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right)V(x_*, x_{k-1}) + \frac{1}{L_k}\Big(f(x_*) - f(x_{k-1})\Big) + \frac{\varepsilon}{2L_k}\right) + \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_k)\Big) + \frac{\varepsilon}{2L_{k+1}} \leqslant \\
\leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right)\left(\left(1 - \frac{\mu}{L_{k-1}}\right)V(x_*, x_{k-2}) + \right.\right. \\
\left.\left. + \frac{1}{L_{k-1}}\Big(f(x_*) - f(x_{k-2})\Big) + \frac{\varepsilon}{2L_{k-1}}\right) + \frac{1}{L_k}\Big(f(x_*) - f(x_{k-1})\Big) + \frac{\varepsilon}{2L_k}\right) + \\
+ \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_k)\Big) + \frac{\varepsilon}{2L_{k+1}} \leqslant \dots \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right) \times \right. \\
\times \left(\left(1 - \frac{\mu}{L_{k-1}}\right)\left(\dots\left(\left(1 - \frac{\mu}{L_1}\right)V(x_*, x_0) + \frac{1}{L_1}\Big(f(x_*) - f(x_0)\Big) + \frac{\varepsilon}{2L_1}\right) + \right.\right. \\
\left.\left.\left. + \frac{1}{L_2}\Big(f(x_*) - f(x_1)\Big) + \frac{\varepsilon}{2L_2}\right) + \dots\right) + \frac{1}{L_k}\Big(f(x_*) - f(x_{k-1})\Big) + \frac{\varepsilon}{2L_k}\right) + \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_k)\Big) + \frac{\varepsilon}{2L_{k+1}}.
\end{gathered}$$

Thus, we have

$$V(x_*, x_{k+1}) \leqslant \sum_{i=1}^{k+1}\left(\frac{\prod\limits_{n=i+1}^{k+1}\left(1 - \frac{\mu}{L_n}\right)}{L_i}\left(f(x_*) - f(x_{i-1}) + \frac{\varepsilon}{2}\right)\right) + \prod_{n=1}^{k+1}\left(1 - \frac{\mu}{L_n}\right)V(x_*, x_0),$$

which implies

$$V(x_*, x_N) \leqslant \sum_{i=1}^{N} \left(\frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \left(f(x_*) - f(x_{i-1}) + \frac{\varepsilon}{2} \right) \right) + \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0). \tag{38}$$

Let $y_N = x_{j(N)}$, where $j(N) = \arg \min_{k=0, \ldots, N-1} f(x_k)$. Using (38), we get

$$\prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0) \geqslant \sum_{i=1}^{N} \left(\frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \left(f(x_{i-1}) - f(x_*) - \frac{\varepsilon}{2} \right) \right) + V(x_*, x_N).$$

Taking into account that $V(x_*, x_N) \geqslant 0$ and $f(y_N) \leqslant f(x_k)$ $\forall k = 0, \ldots, N-1$, we obtain

$$\prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0) \geqslant \sum_{i=1}^{N} \left(\frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \left(f(x_{i-1}) - f(x_*) - \frac{\varepsilon}{2} \right) \right) \geqslant$$

$$\geqslant \sum_{i=1}^{N} \left(\frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \left(f(y_N) - f(x_*) \right) \right) - \frac{\varepsilon}{2} \sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i}.$$

Dividing both sides of the last inequality by

$$\sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i},$$

we have

$$f(y_N) - f(x_*) \leqslant \frac{\prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{\sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i}} V(x_*, x_0) + \frac{\varepsilon}{2}.$$

Let us consider the first case, $L_N > \mu$. Since

$$\prod_{n=i}^{N} \left(1 - \frac{\mu}{L_n}\right) \geqslant \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right)$$

and

$$\sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \geqslant \frac{1}{L_N},$$

the following inequality holds:

$$f(y_N) - f(x_*) \leqslant \min \left\{ L_N \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right), \frac{1}{\sum_{i=1}^{N} \frac{1}{L_i}} \right\} V(x_*, x_0) + \frac{\varepsilon}{2}. \tag{39}$$

Also, since $f(x_*) \leqslant f(x_i)$, using (38), we get

$$V(x_*, x_N) \leqslant \frac{\varepsilon}{2} \sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} + \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0). \tag{40}$$

In the opposite case, if $L_N \leqslant \mu$, then from inequality (37) with $k + 1 = N$ we get

$$V(x_*, x_N) \leqslant \frac{\varepsilon}{2L_N}. \tag{41}$$

Further, from inequality (34) for $x = x_*$, $k = N$ and taking into account that $f(y_N) \leqslant f(x_N)$, we obtain

$$f(y_N) - f(x_*) \leqslant \frac{\varepsilon}{2}. \tag{42}$$

The right-hand sides of the last two inequalities (42) and (41) can be bounded by the right-hand sides of inequalities (39) and (40).

The proof of the estimate for the number of iterations of Algorithm 1 is equivalent to the proof of Theorem 2 given in Appendix B. □

Appendix B. The proof of Theorem 2

For each $k \geqslant 0$, we have

$$\langle \nabla f(x_k), x_k - x \rangle \leqslant L_{k+1} V(x, x_k) - L_{k+1} V(x, x_{k+1}) + \delta_{k+1}.$$

Combining this inequality with

$$\langle \nabla f(x_k), x_k - x \rangle \geqslant f(x_k) - f(x) + \mu V(x, x_k),$$

we have

$$f(x_k) - f(x) + \mu V(x, x_k) \leqslant L_{k+1} V(x, x_k) - L_{k+1} V(x, x_{k+1}) + \delta_{k+1}. \tag{43}$$

After simplifying and combining terms, we rewrite (43) in the following form:

$$L_{k+1} V(x, x_{k+1}) \leqslant \left(L_{k+1} - \mu\right) V(x, x_k) + f(x) - f(x_k) + \delta_{k+1} \quad \forall x \in Q. \tag{44}$$

Further, let us rewrite (44) for $x = x_*$

$$L_{k+1} V(x_*, x_{k+1}) \leqslant (L_{k+1} - \mu) V(x_*, x_k) + f(x_*) - f(x_k) + \delta_{k+1}.$$

Let us divide both sides of the last inequality by L_{k+1}

$$V(x_*, x_{k+1}) \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right) V(x_*, x_k) + \frac{1}{L_{k+1}} \left(f(x_*) - f(x_k)\right) + \frac{\delta_{k+1}}{L_{k+1}}. \tag{45}$$

Continuing inequality (45) recursively, we find that

$$V(x_*, x_{k+1}) \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right) V(x_*, x_k) + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_k)\right) + \frac{\delta_{k+1}}{L_{k+1}} \leqslant$$
$$\leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right) V(x_*, x_{k-1}) + \frac{1}{L_k}\left(f(x_*) - f(x_{k-1})\right) + \frac{\delta_k}{L_k}\right) + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_k)\right) + \frac{\delta_{k+1}}{L_{k+1}} \leqslant$$
$$\leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right)\left(\left(1 - \frac{\mu}{L_{k-1}}\right) V(x_*, x_{k-2}) +\right.\right.$$
$$\left.\left. + \frac{1}{L_{k-1}}\left(f(x_*) - f(x_{k-2})\right) + \frac{\delta_{k+1}}{L_{k-1}}\right) + \frac{1}{L_k}\left(f(x_*) - f(x_{k-1})\right) + \frac{\delta_k}{L_k}\right) +$$
$$+ \frac{1}{L_{k+1}}\left(f(x_*) - f(x_k)\right) + \frac{\delta_{k+1}}{L_{k+1}} \leqslant \ldots \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right)\times\right.$$
$$\times\left(\left(1 - \frac{\mu}{L_{k-1}}\right)\left(\ldots\left(\left(1 - \frac{\mu}{L_1}\right) V(x_*, x_0) + \frac{1}{L_1}\left(f(x_*) - f(x_0)\right) + \frac{\delta_1}{L_1}\right) +\right.\right.$$
$$\left.\left.\left. + \frac{1}{L_2}\left(f(x_*) - f(x_1)\right) + \frac{\delta_2}{L_2}\right) + \ldots\right) + \frac{1}{L_k}\left(f(x_*) - f(x_{k-1})\right) + \frac{\delta_k}{L_k}\right) + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_k)\right) + \frac{\delta_{k+1}}{L_{k+1}}.$$

Thus, we have

$$V(x_*, x_{k+1}) \leqslant \sum_{i=1}^{k+1}\left(\frac{\prod_{n=i+1}^{k+1}\left(1 - \frac{\mu}{L_n}\right)}{L_i}\left(f(x_*) - f(x_{i-1}) + \delta_i\right)\right) + \prod_{n=1}^{k+1}\left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0),$$

which implies

$$V(x_*, x_N) \leqslant \sum_{i=1}^{N}\left(\frac{\prod_{n=i+1}^{N}\left(1 - \frac{\mu}{L_n}\right)}{L_i}\left(f(x_*) - f(x_{i-1}) + \delta_i\right)\right) + \prod_{n=1}^{N}\left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0). \tag{46}$$

Let $y_N = x_{j(N)}$, where $j(N) = \arg\min_{k=0,\ldots,N-1} f(x_k)$. From (46) we get

$$\prod_{n=1}^{N}\left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0) \geqslant \sum_{i=1}^{N}\left(\frac{\prod_{n=i+1}^{N}\left(1 - \frac{\mu}{L_n}\right)}{L_i}\left(f(x_{i-1}) - f(x_*) - \delta_i\right)\right) + V(x_*, x_N).$$

Taking into account that $V(x_*, x_N) \geqslant 0$ and $f(y_N) \leqslant f(x_k)$ $\forall k = 0, \ldots, N-1$, we obtain

$$\prod_{n=1}^{N}\left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0) \geqslant \sum_{i=1}^{N}\left(\frac{\prod_{n=i+1}^{N}\left(1 - \frac{\mu}{L_n}\right)}{L_i}\left(f(x_{i-1}) - f(x_*) - \delta_i\right)\right) \geqslant$$
$$\geqslant \sum_{i=1}^{N}\left(\frac{\prod_{n=i+1}^{N}\left(1 - \frac{\mu}{L_n}\right)}{L_i}\left(f(y_N) - f(x_*)\right)\right) - \sum_{i=1}^{N}\frac{\delta_i \prod_{n=i+1}^{N}\left(1 - \frac{\mu}{L_n}\right)}{L_i}.$$

Dividing both sides of the last inequality by

$$\widehat{S}_N := \sum_{i=1}^{N-1} \frac{\prod\limits_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i} + \frac{1}{L_N},$$

we have

$$f(y_N) - f(x_*) \leqslant \frac{V(x_*, x_0)}{\widehat{S}_N} \prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right) + \frac{1}{\widehat{S}_N} \sum_{i=1}^{N} \frac{\delta_i \prod\limits_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}.$$

Let us consider the first case, $L_N > \mu$. Since

$$\prod_{n=i}^{N}\left(1-\frac{\mu}{L_n}\right) \geqslant \prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right)$$

and

$$\sum_{i=1}^{N} \frac{\prod\limits_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i} \geqslant \frac{1}{L_N},$$

the following inequality holds:

$$f(y_N) - f(x_*) \leqslant L_N \prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right) V(x_*, x_0) + \frac{1}{\widehat{S}_N} \sum_{i=1}^{N} \frac{\delta_i \prod\limits_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}. \tag{47}$$

Also, since $f(x_*) \leqslant f(x_i)$ and using (46), we get

$$V(x_*, x_N) \leqslant \sum_{i=1}^{N} \frac{\delta_i \prod\limits_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i} + \prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right) V(x_*, x_0). \tag{48}$$

Let us prove an estimate for the number of iterations of the algorithm. Due to (13), we have

$$L_N\left(1-\frac{\mu}{L_N}\right)^N V\left(x_*, x_0\right) \leqslant L_N\left(1-\frac{\mu}{L_N}\right)^N R^2 \leqslant \varepsilon$$

if

$$N = \frac{\log \frac{\varepsilon}{L_N R^2}}{\log\left(1-\frac{\mu}{L_N}\right)} \sim \left(\frac{L_N}{\mu} \log \frac{L_N R^2}{\varepsilon}\right).$$

Assuming $L_N \leqslant 2L = O\left(\frac{M^2}{\varepsilon}\right)$, we obtain the following estimate of the number of iterations:

$$N = O\left(\frac{M^2}{\mu\varepsilon} \log \frac{1}{\varepsilon}\right).$$

□

Remark 3. If $L_N \leqslant \mu$, from inequality (45), with $k+1 = N$ we get

$$V(x_*, x_N) \leqslant \delta_N. \tag{49}$$

Further, from inequality (43), for $x = x_*$, $k = N$, and taking into account that $f(y_N) \leqslant f(x_N)$, we obtain

$$f(y_N) - f(x_*) \leqslant \delta_N. \tag{50}$$

The right-hand sides of the last two inequalities (50) and (49) can be bounded by the right-hand sides of inequalities (47) and (48).

Appendix C. The proof of Theorem 3

After the completion of the kth iteration ($k = 0, 1, \ldots$) of Algorithm 3, the following inequalities hold:

$$\langle \nabla f(x_k), x_{k+1} - x_k\rangle \leqslant \langle \nabla f(x_k), x - x_k\rangle + L_{k+1}V(x, x_k) - L_{k+1}V(x, x_{k+1}) - L_{k+1}V(x_{k+1}, x_k),$$

and

$$\langle \nabla f(x_k), x_{k+1} - x_k\rangle \geqslant f(x_{k+1}) - f(x_k) - L_{k+1}V(x_{k+1}, x_k) - \delta_{k+1}.$$

Therefore,

$$f(x_{k+1}) - f(x_k) - L_{k+1}V(x_{k+1}, x_k) - \delta_{k+1} \leqslant \langle \nabla f(x_k), x - x_k\rangle + L_{k+1}V(x, x_k) - \\ - L_{k+1}V(x, x_{k+1}) - L_{k+1}V(x_{k+1}, x_k).$$

Thus, for each $k \geqslant 0$, we have

$$\langle \nabla f(x_k), x_k - x\rangle \leqslant f(x_k) - f(x_{k+1}) + L_{k+1}V(x, x_k) - L_{k+1}V(x, x_{k+1}) + \delta_{k+1}.$$

Combining this inequality with

$$\langle \nabla f(x_k), x_k - x\rangle \geqslant f(x_k) - f(x) + \mu V(x, x_k),$$

we have

$$f(x_k) - f(x) + \mu V(x, x_k) \leqslant f(x_k) - f(x_{k+1}) + L_{k+1}V(x, x_k) - L_{k+1}V(x, x_{k+1}) + \delta_{k+1}. \tag{51}$$

After simplifying and combining terms, we rewrite (51) in the following form:

$$L_{k+1}V(x, x_{k+1}) \leqslant (L_{k+1} - \mu)V(x, x_k) + f(x) - f(x_{k+1}) + \delta_{k+1} \quad \forall x \in Q. \tag{52}$$

Further, let us rewrite (52) for $x = x_*$:

$$L_{k+1}V(x_*, x_{k+1}) \leqslant (L_{k+1} - \mu)V(x_*, x_k) + f(x_*) - f(x_{k+1}) + \delta_{k+1}. \tag{53}$$

Dividing both sides of (53) by L_{k+1}, we find:

$$V(x_*, x_{k+1}) \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)V(x_*, x_k) + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_{k+1})\right) + \frac{\delta_{k+1}}{L_{k+1}}. \tag{54}$$

Continuing inequality (54) recursively, we obtain

$$V(x_*, x_{k+1}) \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)V(x_*, x_k) + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_{k+1})\right) + \frac{\delta_{k+1}}{L_{k+1}} \leqslant \\ \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right)V(x_*, x_{k-1}) + \frac{1}{L_k}\left(f(x_*) - f(x_k)\right) + \frac{\delta_k}{L_k}\right) + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_{k+1})\right) + \frac{\delta_{k+1}}{L_{k+1}} \leqslant \\ \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right)\left(\left(1 - \frac{\mu}{L_{k-1}}\right)V(x_*, x_{k-2}) + \right.\right. \\ \left.\left. + \frac{1}{L_{k-1}}\left(f(x_*) - f(x_{k-1})\right) + \frac{\delta_{k+1}}{L_{k-1}}\right) + \frac{1}{L_k}\left(f(x_*) - f(x_k)\right) + \frac{\delta_k}{L_k}\right) + \\ + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_{k+1})\right) + \frac{\delta_{k+1}}{L_{k+1}} \leqslant \ldots \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right) \times \right. \\ \times \left(\left(1 - \frac{\mu}{L_{k-1}}\right)\left(\ldots\left(\left(1 - \frac{\mu}{L_1}\right)V(x_*, x_0) + \frac{1}{L_1}\left(f(x_*) - f(x_1)\right) + \frac{\delta_1}{L_1}\right) + \right.\right. \\ \left.\left. + \frac{1}{L_2}\left(f(x_*) - f(x_2)\right) + \frac{\delta_2}{L_2}\right) + \ldots\right) + \frac{1}{L_k}\left(f(x_*) - f(x_k)\right) + \frac{\delta_k}{L_k}\right) + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_{k+1})\right) + \frac{\delta_{k+1}}{L_{k+1}}.$$

Thus, we have

$$V(x_*, x_{k+1}) \leqslant \sum_{i=1}^{k+1} \left(\frac{\prod\limits_{n=i+1}^{k+1} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \Big(f(x_*) - f(x_i) + \delta_i\Big) \right) + \prod_{n=1}^{k+1} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0),$$

which implies

$$V(x_*, x_N) \leqslant \sum_{i=1}^{N} \left(\frac{\prod\limits_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \Big(f(x_*) - f(x_i) + \delta_i\Big) \right) + \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0). \tag{55}$$

Let $y_N = x_{j(N)}$, where $j(N) = \arg \min\limits_{k=1, \ldots, N} f(x_k)$. From (55) we get

$$\prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0) \geqslant \sum_{i=1}^{N} \left(\frac{\prod\limits_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \Big(f(x_i) - f(x_*) - \delta_i\Big) \right) + V(x_*, x_N).$$

Taking into account that $V(x_*, x_N) \geqslant 0$ and $f(y_N) \leqslant f(x_k)$ $\forall k = 1, \ldots, N$, we obtain

$$\prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0) \geqslant \sum_{i=1}^{N} \left(\frac{\prod\limits_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \Big(f(x_i) - f(x_*) - \delta_i\Big) \right) \geqslant$$

$$\geqslant \sum_{i=1}^{N} \left(\frac{\prod\limits_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \Big(f(y_N) - f(x_*)\Big) \right) - \sum_{i=1}^{N} \frac{\delta_i \prod\limits_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i}.$$

Dividing both sides of the last inequality by

$$\widehat{S}_N := \sum_{i=1}^{N-1} \frac{\prod\limits_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} + \frac{1}{L_N},$$

we have

$$f(y_N) - f(x_*) \leqslant \frac{V(x_*, x_0)}{\widehat{S}_N} \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) + \frac{1}{\widehat{S}_N} \sum_{i=1}^{N} \frac{\delta_i \prod\limits_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i}.$$

Let us consider the first case, $L_N \geqslant \mu$. Since

$$\prod_{n=i}^{N} \left(1 - \frac{\mu}{L_n}\right) \geqslant \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right)$$

and

$$\sum_{i=1}^{N} \frac{\prod\limits_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \geqslant \frac{1}{L_N},$$

the following inequality holds:

$$f(y_N) - f(x_*) \leqslant L_N \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0) + \frac{1}{\widehat{S}_N} \sum_{i=1}^{N} \frac{\delta_i \prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i}. \tag{56}$$

Also, since $f(x_*) \leqslant f(x_i)$ and using (55), we get

$$V(x_*, x_N) \leqslant \sum_{i=1}^{N} \frac{\delta_i \prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} + \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0). \tag{57}$$

Let us prove an estimate for the number of iterations of the algorithm considered. Due to (17), we have

$$L_N \left(1 - \frac{\mu}{L_N}\right)^N V\left(x_*, x_0\right) \leqslant L_N \left(1 - \frac{\mu}{L_N}\right)^N R^2 \leqslant \varepsilon$$

if

$$N = \frac{\log \frac{\varepsilon}{L_N R^2}}{\log\left(1 - \frac{\mu}{L_N}\right)} \sim \left(\frac{L_N}{\mu} \log \frac{L_N R^2}{\varepsilon}\right).$$

Assuming $L_N \leqslant 2L = O\left(\frac{M^2}{\varepsilon}\right)$, we obtain the following estimate of the number of iterations:

$$N = O\left(\frac{M^2}{\mu\varepsilon} \log \frac{1}{\varepsilon}\right).$$

□

REMARK 4. If $L_N \leqslant \mu$, from inequality (54), with $k + 1 = N$, we get

$$V(x_*, x_N) \leqslant \delta_N. \tag{58}$$

Further, from inequality (51), for $x = x_*$, $k + 1 = N$, and taking into account that $f(y_N) \leqslant f(x_N)$, we obtain

$$f(y_N) - f(x_*) \leqslant \delta_N. \tag{59}$$

The right-hand sides of the last two inequalities (59) and (58) can be bounded by the right-hand sides of inequalities (56) and (57).

Appendix D. The proof of Theorem 4

After the completion of the kth iteration ($k = 0, 1, \ldots$) of Algorithm 4, the following inequalities hold:

$$\langle \nabla f(x_k), x_{k+1} - x_k \rangle \leqslant \langle \nabla f(x_k), x - x_k \rangle + L_{k+1} V(x, x_k) - L_{k+1} V(x, x_{k+1}) - L_{k+1} V(x_{k+1}, x_k),$$

and

$$\langle \nabla f(x_k), x_{k+1} - x_k \rangle \geqslant f(x_{k+1}) - f(x_k) - L_{k+1} V(x_{k+1}, x_k) - \frac{3\varepsilon}{4}.$$

Therefore,

$$f(x_{k+1}) - f(x_k) - L_{k+1} V(x_{k+1}, x_k) - \frac{3\varepsilon}{4} \leqslant \langle \nabla f(x_k), x - x_k \rangle + L_{k+1} V(x, x_k) - \\ - L_{k+1} V(x, x_{k+1}) - L_{k+1} V(x_{k+1}, x_k).$$

Thus, for each $k \geqslant 0$, we have

$$\langle \nabla f(x_k),\, x_k - x\rangle \leqslant f(x_k) - f(x_{k+1}) + L_{k+1} V(x,\, x_k) - L_{k+1} V(x,\, x_{k+1}) + \frac{3\varepsilon}{4}.$$

Combining this inequality with

$$\langle \nabla f(x_k),\, x_k - x\rangle \geqslant f(x_k) - f(x) + \mu V(x,\, x_k),$$

we have

$$f(x_k) - f(x) + \mu V(x,\, x_k) \leqslant f(x_k) - f(x_{k+1}) + L_{k+1} V(x,\, x_k) - L_{k+1} V(x,\, x_{k+1}) + \frac{3\varepsilon}{4}. \tag{60}$$

After simplifying and combining terms, we rewrite (60) in the following form:

$$L_{k+1} V(x,\, x_{k+1}) \leqslant \left(L_{k+1} - \mu\right) V(x, x_k) + f(x) - f(x_{k+1}) + \frac{3\varepsilon}{4} \quad \forall x \in Q. \tag{61}$$

Further, let us rewrite (61) for $x = x_*$ as

$$L_{k+1} V(x_*,\, x_{k+1}) \leqslant (L_{k+1} - \mu) V(x_*,\, x_k) + f(x_*) - f(x_{k+1}) + \frac{3\varepsilon}{4}. \tag{62}$$

By dividing both sides of inequality (62) by L_{k+1}, we have

$$V(x_*,\, x_{k+1}) \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right) V(x_*,\, x_k) + \frac{1}{L_{k+1}} \left(f(x_*) - f(x_{k+1})\right) + \frac{3\varepsilon}{4L_{k+1}}. \tag{63}$$

Continuing inequality (63) recursively, we obtain

$$\begin{aligned}
V(x_*,\, x_{k+1}) &\leqslant \left(1 - \frac{\mu}{L_{k+1}}\right) V(x_*,\, x_k) + \frac{1}{L_{k+1}} \left(f(x_*) - f(x_{k+1})\right) + \frac{3\varepsilon}{4L_{k+1}} \leqslant \\
&\leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right) V(x_*, x_{k-1}) + \frac{1}{L_k}\left(f(x_*) - f(x_k)\right) + \frac{3\varepsilon}{4L_k}\right) + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_{k+1})\right) + \frac{3\varepsilon}{4L_{k+1}} \leqslant \\
&\leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right)\left(\left(1 - \frac{\mu}{L_{k-1}}\right) V(x_*,\, x_{k-2}) + \right.\right.\\
&\left.\left. + \frac{1}{L_{k-1}}\left(f(x_*) - f(x_{k-1})\right) + \frac{\varepsilon}{2L_{k-1}}\right) + \frac{1}{L_k}\left(f(x_*) - f(x_k)\right) + \frac{3\varepsilon}{4L_k}\right) + \\
&+ \frac{1}{L_{k+1}}\left(f(x_*) - f(x_{k+1})\right) + \frac{3\varepsilon}{4L_{k+1}} \leqslant \ldots \leqslant \left(1 - \frac{\mu}{L_{k+1}}\right)\left(\left(1 - \frac{\mu}{L_k}\right) \times \right.\\
&\times \left(\left(1 - \frac{\mu}{L_{k-1}}\right)\left(\ldots\left(\left(1 - \frac{\mu}{L_1}\right) V(x_*,\, x_0) + \frac{1}{L_1}\left(f(x_*) - f(x_1)\right) + \frac{3\varepsilon}{4L_1}\right) + \right.\right.\\
&\left.\left.\left. + \frac{1}{L_2}\left(f(x_*) - f(x_2)\right) + \frac{3\varepsilon}{4L_2}\right) + \ldots\right) + \frac{1}{L_k}\left(f(x_*) - f(x_k)\right) + \frac{3\varepsilon}{4L_k}\right) + \frac{1}{L_{k+1}}\left(f(x_*) - f(x_{k+1})\right) + \frac{3\varepsilon}{4L_{k+1}}.
\end{aligned}$$

Thus, we have

$$V(x_*,\, x_{k+1}) \leqslant \sum_{i=1}^{k+1} \left(\frac{\prod\limits_{n=i+1}^{k+1} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \left(f(x_*) - f(x_i) + \frac{3\varepsilon}{4}\right) \right) + \prod_{n=1}^{k+1} \left(1 - \frac{\mu}{L_n}\right) V(x_*,\, x_0),$$

which implies

$$V(x_*, x_N) \leqslant \sum_{i=1}^{N} \left(\frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \left(f(x_*) - f(x_i) + \frac{3\varepsilon}{4} \right) \right) + \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0). \tag{64}$$

Let $y_N = x_{j(N)}$, where $j(N) = \arg \min_{k=1,\ldots,N} f(x_k)$. Using (64), we get

$$\prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0) \geqslant \sum_{i=1}^{N} \left(\frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \left(f(x_i) - f(x_*) - \frac{3\varepsilon}{4} \right) \right) + V(x_*, x_N).$$

Taking into account that $V(x_*, x_N) \geqslant 0$ and $f(y_N) \leqslant f(x_k)$ $\forall k = 1, \ldots, N$, we obtain

$$\prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0) \geqslant \sum_{i=1}^{N} \left(\frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \left(f(x_i) - f(x_*) - \frac{3\varepsilon}{4} \right) \right) \geqslant$$

$$\geqslant \sum_{i=1}^{N} \left(\frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \left(f(y_N) - f(x_*) \right) \right) - \frac{3\varepsilon}{4} \sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i}.$$

Dividing both sides of the last inequality by

$$\sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i},$$

we have

$$f(y_N) - f(x_*) \leqslant \frac{\prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{\sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i}} V(x_*, x_0) + \frac{3\varepsilon}{4}.$$

Let us consider the first case, $L_N > \mu$.

Since

$$\prod_{n=i}^{N} \left(1 - \frac{\mu}{L_n}\right) \geqslant \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right)$$

and

$$\sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} \geqslant \frac{1}{L_N},$$

the following inequality holds:

$$f(y_N) - f(x_*) \leqslant \min \left\{ L_N \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right), \frac{1}{\sum_{i=1}^{N} \frac{1}{L_i}} \right\} V(x_*, x_0) + \frac{3\varepsilon}{4}. \tag{65}$$

Also, since $f(x_*) \leqslant f(x_i)$, using (64), we get

$$V(x_*, x_N) \leqslant \frac{3\varepsilon}{4} \sum_{i=1}^{N} \frac{\prod_{n=i+1}^{N} \left(1 - \frac{\mu}{L_n}\right)}{L_i} + \prod_{n=1}^{N} \left(1 - \frac{\mu}{L_n}\right) V(x_*, x_0). \quad (66)$$

In the opposite case, if $L_N \leqslant \mu$, then from inequality (63) with $k + 1 = N$ we get

$$V(x_*, x_N) \leqslant \frac{3\varepsilon}{4L_N}. \quad (67)$$

Further, from inequality (60) for $x = x_*$, $k + 1 = N$ and taking into account that $f(y_N) \leqslant f(x_N)$, we obtain

$$f(y_N) - f(x_*) \leqslant \frac{3\varepsilon}{4}. \quad (68)$$

The right-hand sides of the last two inequalities (68) and (67) can be bounded by the right-hand sides of inequalities (65) and (66).

The proof of the estimate for the number of iterations of the algorithm is equivalent to the proof of Theorem 3 given in Appendix C. □

Appendix E. The proof of Theorem 5

After the completion of the kth iteration ($k = 0, 1, \ldots$) of Algorithm 5, the following inequalities hold:

$$\langle g_\delta(x_k), x_{k+1} - x_k\rangle \leqslant \langle g_\delta(x_k), x - x_k\rangle + L_{k+1}V(x, x_k) - L_{k+1}V(x, x_{k+1}) - L_{k+1}V(x_{k+1}, x_k),$$

and

$$\langle g_\delta(x_k), x_{k+1} - x_k\rangle \geqslant f_\delta(x_{k+1}) - f_\delta(x_k) - L_{k+1}V(x_{k+1}, x_k) - \delta_{k+1}.$$

Since

$$f(x) \leqslant f_\delta(x) + \delta \Longrightarrow f_\delta(x) \geqslant f(x) - \delta \quad \forall x \in Q,$$

we obtain

$$\langle g_\delta(x_k), x_{k+1} - x_k\rangle \geqslant f(x_{k+1}) - f_\delta(x_k) - L_{k+1}V(x_{k+1}, x_k) - \delta - \delta_{k+1}.$$

Therefore,

$$f(x_{k+1}) - f_\delta(x_k) - L_{k+1}V(x_{k+1}, x_k) - \delta - \delta_{k+1} \leqslant \langle g_\delta(x_k), x - x_k\rangle + L_{k+1}V(x, x_k) - \\ - L_{k+1}V(x, x_{k+1}) - L_{k+1}V(x_{k+1}, x_k).$$

Thus, for each $k \geqslant 0$, we have

$$\langle g_\delta(x_k), x_k - x\rangle \leqslant f_\delta(x_k) - f(x_{k+1}) + L_{k+1}V(x, x_k) - L_{k+1}V(x, x_{k+1}) + \delta + \delta_{k+1}. \quad (69)$$

Taking into account the inequality

$$\langle g_\delta(x_k), x - x_k\rangle \geqslant f_\delta(x_k) - f(x) + \mu V(x, x_k)$$

for $x = x_*$ we obtain

$$f(x_{k+1}) - f(x_*) \leqslant L_{k+1}V(x_*, x_k) - L_{k+1}V(x_*, x_{k+1}) - \mu V(x_*, x_k) + \delta + \delta_{k+1}. \quad (70)$$

After simplifying and combining terms, we rewrite (70) in the following form:

$$L_{k+1}V(x_*, x_{k+1}) \leqslant (L_{k+1}-\mu)V(x_*, x_k) + f(x_*) - f(x_{k+1}) + \delta + \delta_{k+1}. \tag{71}$$

Let us divide both sides of the last inequality by L_{k+1}

$$V(x_*, x_{k+1}) \leqslant \left(1-\frac{\mu}{L_{k+1}}\right)V(x_*, x_k) + \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_{k+1})\Big) + \frac{\delta+\delta_{k+1}}{L_{k+1}}. \tag{72}$$

If $L_{k+1} \leqslant \mu$, from inequality (72) we get

$$V(x_*, x_{k+1}) \leqslant \delta + \delta_{k+1}. \tag{73}$$

Further, from inequality (70) we obtain

$$f(x_{k+1}) - f(x_*) \leqslant \delta + \delta_{k+1}. \tag{74}$$

Therefore, we consider the case $L_{k+1} \geqslant \mu$. Continuing inequality (72) recursively, we obtain

$$\begin{aligned}
V(x_*, x_{k+1}) &\leqslant \left(1-\frac{\mu}{L_{k+1}}\right)V(x_*, x_k) + \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_{k+1})\Big) + \frac{\delta+\delta_{k+1}}{L_{k+1}} \leqslant \\
&\leqslant \left(1-\frac{\mu}{L_{k+1}}\right)\left(\left(1-\frac{\mu}{L_k}\right)V(x_*, x_{k-1}) + \frac{1}{L_k}\Big(f(x_*) - f(x_k)\Big) + \frac{\delta_k+\delta}{L_k}\right) + \\
&\quad + \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_{k+1})\Big) + \frac{\delta_{k+1}+\delta}{L_{k+1}} \leqslant \\
&\leqslant \left(1-\frac{\mu}{L_{k+1}}\right)\left(\left(1-\frac{\mu}{L_k}\right)\left(\left(1-\frac{\mu}{L_{k-1}}\right)V(x_*, x_{k-2}) + \right.\right. \\
&\quad \left.\left. + \frac{1}{L_{k-1}}\Big(f(x_*) - f(x_{k-1})\Big) + \frac{\delta_{k+1}+\delta}{L_{k-1}}\right) + \frac{1}{L_k}\Big(f(x_*) - f(x_k)\Big) + \frac{\delta_k+\delta}{L_k}\right) + \\
&\quad + \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_{k+1})\Big) + \frac{\delta_{k+1}+\delta}{L_{k+1}} \leqslant \ldots \leqslant \left(1-\frac{\mu}{L_{k+1}}\right)\left(\left(1-\frac{\mu}{L_k}\right)\times\right. \\
&\quad \times\left(\left(1-\frac{\mu}{L_{k-1}}\right)\ldots\left(\left(1-\frac{\mu}{L_1}\right)V(x_*, x_0) + \frac{1}{L_1}\Big(f(x_*) - f(x_1)\Big) + \frac{\delta_1+\delta}{L_1}\right) + \right. \\
&\quad \left.\left. + \frac{1}{L_2}\Big(f(x_*) - f(x_2)\Big) + \frac{\delta_2+\delta}{L_2}\right) + \ldots\right) + \frac{1}{L_k}\Big(f(x_*) - f(x_k)\Big) + \frac{\delta_k+\delta}{L_k}\right) + \\
&\quad + \frac{1}{L_{k+1}}\Big(f(x_*) - f(x_{k+1})\Big) + \frac{\delta_{k+1}+\delta}{L_{k+1}}.
\end{aligned}$$

Thus, we have

$$V(x_*, x_{k+1}) \leqslant \sum_{i=1}^{k+1}\left(\frac{\prod\limits_{n=i+1}^{k+1}\left(1-\frac{\mu}{L_n}\right)}{L_i}\Big(f(x_*) - f(x_i) + \delta_i + \delta\Big)\right) + \prod_{n=1}^{k+1}\left(1-\frac{\mu}{L_n}\right)V(x_*, x_0),$$

which implies

$$V(x_*, x_N) \leqslant \sum_{i=1}^{N}\left(\frac{\prod\limits_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}\Big(f(x_*) - f(x_i) + \delta_i + \delta\Big)\right) + \prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right)V(x_*, x_0). \tag{75}$$

Let $y_N = x_{j(N)}$, where $j(N) = \arg\min_{k=1,\ldots,N} f(x_k)$. From (75) we get

$$\prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right)V(x_*, x_0) \geqslant \sum_{i=1}^{N}\left(\frac{\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}\Big(f(x_i)-f(x_*)-\delta_i-\delta\Big)\right)+V(x_*, x_N).$$

Taking into account that $V(x_*, x_N) \geqslant 0$ and $f(y_N) \leqslant f(x_k)$, we obtain

$$\prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right)V(x_*, x_0) \geqslant \sum_{i=1}^{N}\left(\frac{\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}\Big(f(x_i)-f(x_*)-\delta_i-\delta\Big)\right) \geqslant$$

$$\geqslant \sum_{i=1}^{N}\left(\frac{\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}\Big(f(y_N)-f(x_*)\Big)\right)-\delta\sum_{i=1}^{N}\frac{\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}-\sum_{i=1}^{N}\frac{\delta_i\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}.$$

Dividing both sides of the last inequality by

$$\widehat{S}_N := \sum_{i=1}^{N-1}\frac{\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}+\frac{1}{L_N},$$

we have

$$f(y_N)-f(x_*) \leqslant \frac{V(x_*, x_0)}{\widehat{S}_N}\prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right)+\frac{\delta}{\widehat{S}_N}\sum_{i=1}^{N}\frac{\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}+\frac{1}{\widehat{S}_N}\sum_{i=1}^{N}\frac{\delta_i\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}.$$

Since

$$\prod_{n=i}^{N}\left(1-\frac{\mu}{L_n}\right) \geqslant \prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right)$$

and

$$\sum_{i=1}^{N}\frac{\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i} \geqslant \frac{1}{L_N},$$

the following inequality holds:

$$f(y_N)-f(x_*) \leqslant L_N\prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right)V(x_*, x_0)+\frac{\delta}{\widehat{S}_N}\sum_{i=1}^{N}\frac{\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}+\frac{1}{\widehat{S}_N}\sum_{i=1}^{N}\frac{\delta_i\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}. \quad (76)$$

Also, since $f(x_*) \leqslant f(x_i)$ and using (75), we get

$$V(x_*, x_N) \leqslant \sum_{i=1}^{N}\frac{\delta_i\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}+\delta\sum_{i=1}^{N}\frac{\prod_{n=i+1}^{N}\left(1-\frac{\mu}{L_n}\right)}{L_i}+\prod_{n=1}^{N}\left(1-\frac{\mu}{L_n}\right)V(x_*, x_0). \quad (77)$$

The right-hand sides of the last two inequalities (74) and (73) can be bounded by the right-hand sides of inequalities (76) and (77). □

Appendix F. The proof of Theorem 6

First of all, we need to prove the following lemma.

Lemma 1. *Let f and g be relatively strongly convex functions, and let y be an ε-solution*

$$f(y) - f(x_*) \leqslant \varepsilon, \quad g(y) \leqslant \varepsilon$$

of the problem (26)–(27)*. Then $\mu V(y, x_*) \leqslant \varepsilon$.*

It is well known that, due to the necessary optimality condition, there exist constants nonzero at the same time, $\lambda_1, \lambda_2 \geqslant 0$ and subgradients $\nabla f(x_*)$, $\nabla g(x_*)$ such that the following inequality takes place:

$$\langle \lambda_1 \nabla f(x_*) + \lambda_2 \nabla g(x_*), x - x_* \rangle \geqslant 0 \ \forall x \in Q, \quad \text{and} \quad \lambda_2 g(x_*) = 0. \tag{78}$$

Let us consider all possible cases of λ_1, λ_2 in (78)

1. $\lambda_1 = 0, \lambda_2 > 0$. According to (78), we have $\langle \nabla g(x_*), x - x_* \rangle \geqslant 0 \ \forall x \in Q$, so $\langle \nabla g(x_*), y - x_* \rangle \geqslant 0$. Recall that $g(x_*) = 0$. Therefore,

$$g(y) \geqslant g(x_*) + \langle \nabla g(x_*), y - x_* \rangle + \mu V(x_*, y) \geqslant \mu V(x_*, y).$$

As y is an ε-solution of the problem, we get $g(y) \leqslant \varepsilon$, which confirms the statement of the lemma.

2. $\lambda_1 > 0, \lambda_2 = 0$. Similarly, according to (78), we have $\langle \nabla f(x_*), x - x_* \rangle \geqslant 0 \ \forall x \in Q$, so $\langle \nabla f(x_*), y - x_* \rangle \geqslant 0$. Therefore,

$$f(y) \geqslant f(x_*) + \langle \nabla f(x_*), y - x_* \rangle + \mu V(x_*, y) \geqslant f(x_*) + \mu V(x_*, y).$$

As y is an ε-solution of the problem, we find that $f(y) - f(x_*) \leqslant \varepsilon$, so

$$\mu V(x_*, y) \leqslant \varepsilon.$$

3. $\lambda_1 > 0, \lambda_2 > 0$. For this case we have

$$\lambda_1 \langle \nabla f(x_*), x - x_* \rangle + \lambda_2 \langle \nabla g(x_*), x - x_* \rangle \geqslant 0 \quad \forall x \in Q.$$

So, either $\langle \nabla f(x_*), x - x_* \rangle \geqslant 0 \ \forall x \in Q$ and the proof is similar to item 2, or $\langle \nabla g(x_*), x - x_* \rangle \geqslant 0$ and the proof is identical to item 1. Thus, the lemma is proved.

Let us now prove the statement of Theorem 6. Using induction let us show that $\forall p \in \mathbb{N}$: $V(x_p, x_*) \leqslant R_p^2$.

First of all, according to the assumption, we have $V(x_0, x_*) \leqslant R_0^2$. Now, let us assume that for some $p > 0$ $V(x_p, x_*) \leqslant R_p^2$. Using the theorem concerning the rate of convergence of Algorithm 6, one can find that after no more than

$$k_{p+1} = 1 + \left\lceil \frac{\Omega M^2 R_p^2}{\varepsilon_{p+1}^2} \right\rceil$$

iterations, for the corresponding x_{p+1}, the following inequalities hold:

$$f(x_{p+1}) - f(x_*) \leqslant \varepsilon_{p+1}, \quad \text{and} \quad g(x_{p+1}) \leqslant \varepsilon_{p+1}.$$

Recall that $M = \max\{M_f, M_g\}$. According to Lemma (1), for the ε_{p+1}-solution, the following inequality holds:

$$V(x_{p+1}, x_*) \leqslant \frac{\varepsilon_{p+1}}{\mu} = R_{p+1}^2.$$

Thus, we have proved that

$$V(x_p, x_*) \leqslant R_p^2 \quad \forall p \in \mathbb{N}.$$

Note that at the same time

$$f(x_p) - f(x_*) \leqslant \varepsilon_p = \mu R_p^2 = \mu R_0^2 \cdot 2^{-p},$$

and

$$g(x_p) \leqslant \varepsilon_p = \mu R_p^2 = \mu R_0^2 \cdot 2^{-p}.$$

Thus, for $p > \log_2 \frac{\mu R_0^2}{\varepsilon}$, the point x_p will be an ε-solution of the problem and $V(x_p, x_*) \leqslant \frac{\varepsilon}{\mu}$.

Let us denote $\widetilde{p} = \log_2 \frac{\mu R_0^2}{\varepsilon}$. The total number of iterations of Algorithm 7 will be

$$N = \sum_{p=0}^{\widetilde{p}} k_p \leqslant \sum_{p=0}^{\widetilde{p}} \left(1 + \frac{\Omega M^2 R_p^2}{\varepsilon_p^2}\right) \leqslant \sum_{p=0}^{\widetilde{p}} \left(1 + \frac{\Omega M^2 R_p^2}{\mu^2 R_p^4}\right) \leqslant$$

$$\leqslant \sum_{p=0}^{\widetilde{p}} \left(1 + \frac{\Omega M^2 2^p}{\mu^2 R_0^2}\right) \leqslant \widetilde{p} + \frac{\Omega M^2 2^{\widetilde{p}}}{\mu^2 R_0^2} = \widetilde{p} + \frac{\Omega M^2 \mu R_0^2}{\varepsilon \mu^2 R_0^2 - 2\delta \mu^2 R_0^2} = \widetilde{p} + \frac{\Omega M^2}{\mu \varepsilon} = \widetilde{p} + \frac{\Omega M^2}{\mu \varepsilon}.$$