mathematics

MDPI

Article

Newtonian Property of Subgradient Method with Optimization of Metric Matrix Parameter Correction

Elena Tovbis [1], Vladimir Krutikov [1,2] and Lev Kazakovtsev [1,*]

[1] Institute of Informatics and Telecommunications, Reshetnev Siberian State University of Science and Technology, 31, Krasnoyarskii Rabochii Prospekt, Krasnoyarsk 660037, Russia; sibstu2006@rambler.ru (E.T.); krutikovvn@rambler.ru (V.K.)
[2] Department of Applied Mathematics, Kemerovo State University, 6 Krasnaya Street, Kemerovo 650043, Russia
* Correspondence: levk@bk.ru

Abstract: The work proves that under conditions of instability of the second derivatives of the function in the minimization region, the estimate of the convergence rate of Newton's method is determined by the parameters of the irreducible part of the conditionality degree of the problem. These parameters represent the degree of difference between eigenvalues of the matrices of the second derivatives in the coordinate system, where this difference is minimal, and the resulting estimate of the convergence rate subsequently acts as a standard. The paper studies the convergence rate of the relaxation subgradient method (RSM) with optimization of the parameters of two-rank correction of metric matrices on smooth strongly convex functions with a Lipschitz gradient without assumptions about the existence of second derivatives of the function. The considered RSM is similar in structure to quasi-Newton minimization methods. Unlike the latter, its metric matrix is not an approximation of the inverse matrix of second derivatives but is adjusted in such a way that it enables one to find the descent direction that takes the method beyond a certain neighborhood of the current minimum as a result of one-dimensional minimization along it. This means that the metric matrix enables one to turn the current gradient into a direction that is gradient-consistent with the set of gradients of some neighborhood of the current minimum. Under broad assumptions on the parameters of transformations of metric matrices, an estimate of the convergence rate of the studied RSM and an estimate of its ability to exclude removable linear background are obtained. The obtained estimates turn out to be qualitatively similar to estimates for Newton's method. In this case, the assumption of the existence of second derivatives of the function is not required. A computational experiment was carried out in which the quasi-Newton BFGS method and the subgradient method under study were compared on various types of smooth functions. The testing results indicate the effectiveness of the subgradient method in minimizing smooth functions with a high degree of conditionality of the problem and its ability to eliminate the linear background that worsens the convergence.

Keywords: minimization; subgradient method; convergence rate

MSC: 90C53

Citation: Tovbis, E.; Krutikov, V.; Kazakovtsev, L. Newtonian Property of Subgradient Method with Optimization of Metric Matrix Parameter Correction. *Mathematics* **2024**, *12*, 1618. https://doi.org/10.3390/math12111618

Academic Editor: Alicia Cordero

Received: 30 March 2024
Revised: 14 May 2024
Accepted: 16 May 2024
Published: 22 May 2024

1. Introduction

We consider a problem of minimization of convex differentiable function $f(x)$, $x \in R^n$, where R^n is finite-dimensional Euclidean space. Under conditions of a high degree of a function degeneracy, it is necessary to use Newton-type minimization methods, for example, modifications of Newton's method, or quasi-Newton methods.

While in the minimum neighborhood, there is a stable quadratic representation of the function, most iterations of the minimization method take place outside the extremum area, so it seems relevant to study the accelerating properties of methods with changing the space metric under conditions of instability of the quadratic properties of the function.

Numerous studies on the convergence rate of Newton methods and quasi-Newton methods in the extremum region have been conducted, and some of them are given in [1–8]. The results obtained in [9,10] refer to the convergence rate of quasi-Newton minimization methods under the assumption that the method operates in the extremum area of the function. The authors of [11] aimed at accelerating the symmetric rank-1 quasi-Newton method with Nesterov's gradient. A convergence rate of incremental quasi-Newton method was investigated in [12,13]. Large-scale optimization through the sampled versions of quasi-Newton method was considered in [14,15]. Also, the convergence rate of randomized and greedy variants of Newtonian methods and quasi-Newton methods were presented in [16–24].

As an object of minimization, we use strongly convex functions with a Lipschitz gradient [25]. In the case of the existence of second derivatives, these constants limit the spread of the Hessian eigenvalues in the minimization region [25]. The ratio $\rho/L \leq 1$ of the strong convexity constant ρ and Lipschitz constant L determines the convergence rate of gradient minimization methods with an indicator $q \approx 1 - \rho/L$ of approach to the extremum by function [26].

As the presence of a removable linear background, we will understand the existence of a linear coordinate transformation $V \in R^{nxn}$ that allows us to significantly increase the ratio of constants in the new coordinate system $\rho_V/L_V >> \rho/L$. The advantages of the gradient method in the new coordinate system with the indicator $q \approx 1 - \rho_V/L_V$ are obvious. However, this estimate is not feasible, since the transformation V is not known.

This research is a continuation of previous studies [27,28] and aimed at studying the capabilities of the Newton's method and the relaxation subgradient method with optimization of the parameters of rank-two correction of metric matrices [27] to eliminate the linear background that worsens the convergence in the conditions of the existence of transformation V with the properties noted above. Similar studies for quasi-Newton methods were carried out in [29].

Newton's method is invariant with respect to linear coordinate transformation and allows one to obtain an estimate of the convergence rate for Newton's method with indicator $q \approx 1 - \rho^2{}_V/L^2{}_V$. This makes it possible to draw a conclusion about the ability of Newton's method to exclude from the function being minimized a linear background that worsens the convergence, which is eliminated using a linear transformation of coordinates. In what follows, this estimate serves as a standard, and the ability of a certain method, like Newton's method, to exclude linear background will be called its Newtonian property. The main goal of the work is to substantiate the presence of the Newtonian property in RSM with a change in the space metric [27]. As shown in [29], the noted Newtonian property is inherent in quasi-Newton methods.

There are a number of directions for constructing non-smooth optimization methods, some of which are given in [25,30,31]. The works [32–34] considered an approach to creating smooth approximations for non-smooth functions. Methods of this class are applicable to a wide range of problems. A number of effective approaches in the field of non-smooth optimization arose as a result of the creation of the first subgradient methods with space dilation [35,36], in the class of minimization methods relaxing both in function and in distance to the extremum [25,37,38].

The first RSMs were proposed in [39–41]. In [36], an effective RSM with space dilation in the direction of the subgradient difference (RSMSD) was developed. Subsequent work on the creation of effective RSMs is associated with identifying the origin of RSMSD and its theoretical justification [42,43]. Formalization of the model of subgradient sets and the use of ideas and machine learning algorithms [44] made it possible to identify the principles of organizing RSM with space dilation [43] and obtain a theoretical basis for their creation. It turned out that the problem of finding the descent direction in RSM can be reduced to the problem of solving a system of inequalities on subgradient sets and mathematically formulated as a solution to the problem of minimizing a quality functional. In this case,

the convergence rate of the minimization method is determined by the properties of the learning algorithm.

The principle of RSM organizing does not rely on second derivative of functions. The method under study is similar in structure to the quasi-Newton methods, and its formulas for transforming metric matrices are similar in structure to the formulas of the quasi-Newton DFP method. The purpose of converting metric matrices in RSM is to find a metric matrix that transforms subgradients into a direction that forms an acute angle with all subgradients in the neighborhood of the current minimum approximation. Using this direction enables us to go beyond this neighborhood.

Studied RSM with optimization of the parameters of rank-two metric matrix correction [27] is the result of RSM improvement from [43]. The problem of finding the descent direction in the RSM from [43] is reduced to the problem of solving a system of inequalities to develop a descent direction that forms an acute angle with the set of subgradients of a certain neighborhood of the current minimum. In this case, the descent direction is found similarly to how it is done in quasi-Newton methods, by multiplying the matrix by the subgradient. In [27], compared with the algorithm from [43], a faster algorithm for solving systems of inequalities was proposed, which was confirmed by a computational experiment in [27] for RSM on this basis.

In this work, a qualitative analysis of formulas for choosing algorithm parameters from [27] is carried out, and on this basis, a new method is proposed for finding matrix transformation parameters. In contrast to RSM from [27,42], where the algorithm convergence is justified under strict restrictions on the transformation parameters of metric matrices, in this work, estimates of the algorithm convergence rate on smooth functions are obtained for a wide range of matrix transformation parameters. Therefore, one can customize the method to solve problems of a certain class by selecting parameters for converting metric matrices.

For the studied RSM, it is shown that the method is invariant under linear coordinate transformation. An estimate of its convergence rate on strongly convex functions with a Lipschitz gradient is obtained. The property of Newton's method is to eliminate the high degree of conditionality of the minimization problem caused by the linear background, which is also inherent in the subgradient method under study. At the same time, estimates of the convergence rate of Newton's method and the method under study are qualitatively similar in reflecting the influence of the characteristics of the ill-conditioned problem.

To solve both smooth and non-smooth problems, universal algorithms have been developed and implemented, which are the practical implementation of an idealized version of the method. To detect the Newtonian property in the proposed methods, special test functions have been developed. The first of them simulates the random nature of changes in the properties of a function. In another function, a targeted change is made in the elongation of the function level lines along the coordinate axes as it approaches the extremum. In one of the functions, the axes of level lines elongation change due to movement along an ellipsoidal ravine.

In the computational experiment, a comparison is made of the quasi-Newtonian BFGS method and the investigated universal subgradient methods on the proposed test functions. The testing results indicate the effectiveness of the developed methods in minimizing smooth functions with a high degree of conditionality and their ability to exclude linear background that worsens convergence. Depending on the type of function, different methods dominate, which allows us to conclude that the subgradient method is applicable along with quasi-Newton methods in solving problems of minimizing smooth functions with a high degree of conditionality.

The rest of the paper is organized as follows. In Section 2, the accelerating properties of Newton's method under conditions of instability of second derivatives of the function are considered. In Section 3, a subgradient method is presented that solves the problem of forming the direction of descent. The convergence rate of the subgradient method on strongly convex functions with Lipschitz gradient is discussed in Section 4. Features of

the implementation of the subgradient method are presented in Section 5. The results of a numerical study on smooth functions are shown in Section 6. Section 7 concludes the work.

2. Accelerating Properties of Newton's Method under Conditions of Instability of Second Derivatives

Denote $f_k = f(x_k)$. For non-smooth functions, we will denote a vector from the subgradient set $g_k = g(x_k) \in \partial f(x_k)$. Due to the coincidence of the gradient and subgradient on smooth functions, we will also use this notation for smooth functions $g_k = g(x_k) = \nabla f(x_k)$.

Condition 1. *We will assume that the function being minimized f(x), $x \in R^n$ is differentiable and strongly convex in R^n, i.e., there exists $\rho > 0$ such that the inequality:*

$$f(\alpha x + (1-\alpha)y) \le \alpha f(x) + (1-\alpha)f(y) - \alpha(1-\alpha)\rho\|x-y\|^2/2,$$

holds for all $x,y \in R^n$ and $\alpha \in [0, 1]$, and the gradient $g(x) = \nabla f(x)$ satisfies the Lipschitz condition

$$\|g(x) - g(y)\| \le L\|x-y\| \quad \forall x, y \in R^n, \quad L > 0. \tag{1}$$

Functions which fulfill Condition 1 satisfy the relations [25]:

$$f(x) - f^* \le \frac{\|g(x)\|^2}{2\rho}, \ \forall x \in R^n, \tag{2}$$

$$f(x) - f^* \ge \frac{\rho\|x - x^*\|^2}{2}, \ \forall x \in R^n, \tag{3}$$

$$\|g(x)\|^2 \ \le 2L(f(x) - f^*), \ \forall x \in R^n, \tag{4}$$

where x^* is the minimum point and $f^* = f(x^*)$ is the function value at the minimum point.

The iteration of the gradient-consistent method with exact one-dimensional descent has the form:

$$x_{k+1} = x_k - \beta_k s_k, \quad \langle s_k, g(x_k)\rangle > 0, \tag{5}$$

$$\beta_k = \underset{\beta \ge 0}{\operatorname{argmin}} f(x_k - \beta s_k), \tag{6}$$

where the initial point is x_0 and s_k is a search direction.

Theorem 1. *Let the function satisfy Condition 1. Then, the sequence of iterations j = 0, 1,. . ., k of the process (5), (6) is estimated as:*

$$f_{k+1} - f^* \le Q_k(f_0 - f^*), \ Q_k = \textstyle\prod_{j=0}^{k} q_j. \tag{7}$$

$$q_j = 1 - \frac{\rho\langle g_j, s_j\rangle^2}{L\|g_j\|^2\|s_j\|^2} \tag{8}$$

Proof of Theorem 1. We present the exact value of the function reduction indicator ${q_k}^*$ at iteration in the form:

$${q_k}^* = \frac{f_{k+1} - f^*}{f_k - f^*} = \frac{f_{k+1} - f_k + f_k - f^*}{f_k - f^*} = 1 - \frac{f_k - f_{k+1}}{f_k - f^*}. \tag{9}$$

Let us make estimates for numerator and denominator in (9). According to (2), for the denominator, we obtain:

$$f_k - f^* \le \frac{\|g_k\|^2}{2\rho}. \tag{10}$$

According to (6), f_{k+1} is the minimum of a one-dimensional function whose gradient is $\langle g(x_k), s_k\rangle / \|s_k\|$. Due to the fact that this one-dimensional function also satisfies Condition 1, to estimate the numerator in (9), taking into account inequality (4), we obtain:

$$f_k - f_{k+1} \geq \frac{\langle g(x_k), s_k\rangle^2}{2L\|s_k\|^2},$$

Using (9) and (10), we obtain (8):

$$q_k{}^* = 1 - \frac{f_k - f_{k+1}}{f_k - f^*} \leq q_k = 1 - \frac{\rho\langle g_k, s_k\rangle^2}{L\|s_k\|^2\|g_k\|^2}.$$

□

Based on Theorem 1, the convergence rate indicator for the gradient method (5), (6) with a choice of descent direction

$$s_k = \nabla f(x_k) \tag{11}$$

according to (8) and (11), will have the form:

$$q_k = 1 - \frac{\rho}{L}. \tag{12}$$

Let us consider an estimate of the convergence rate of Newton's method under Condition 1 and the assumption of the existence of second derivatives of the function.

Theorem 2. *Let the function be twice differentiable and satisfy Condition 1. Then, for a sequence of iterations j = 0, 1,. . ., k of process (5), (6) with the choice of Newton's method direction*

$$s_k = \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k), \tag{13}$$

the estimation takes place:

$$f_{k+1} - f^* \leq Q_k(f_0 - f^*), \quad Q_k = \prod_{j=0}^{k} q_j, \quad q_j = 1 - \frac{\rho^2}{L^2},\ j = 0, 1, \ldots, k. \tag{14}$$

Proof of Theorem 2. Hessian $\nabla^2 f(x_k)$ under Condition 1 satisfies the constraints [25]:

$$\rho\langle z, z\rangle \leq \left\langle \nabla^2 f(x) z, z \right\rangle \leq L\langle z, z\rangle, \quad \forall z \in R^n. \tag{15}$$

Denote $H_k = \left[\nabla^2 f(x_k)\right]^{-1}$, and ${H_k}^{0.5}$ is a symmetric matrix such that $H_k = {H_k}^{0.5}{H_k}^{0.5}$, $z = {H_k}^{0.5} g_k$.

To use Theorem 1, we estimate $\langle g_k, s_k\rangle^2 / \|g_k\|^2\|s_k\|^2$ for direction (13) subject to constraints (15):

$$\frac{\langle g_k, s_k\rangle^2}{\|g_k\|^2\|s_k\|^2} = \frac{\langle g_k, H_k g_k\rangle^2}{\|g_k\|^2\langle H_k g_k, H_k g_k\rangle} = \frac{\langle z, z\rangle^2}{\left\langle z, {H_k}^{-1} z\right\rangle \langle H_k z, z\rangle} \geq \frac{\rho}{L}.$$

Using the last estimate in (8), we obtain estimate (14). □

Let function $f(x)$ satisfy Condition 1. Define the transformation of variables:

$$\hat{x} = Px, \tag{16}$$

where $P \in R^{n \cdot n}$ is a non-singular matrix. In the new coordinate system, the function to be minimized takes the form:

$$f(x) = f\left(P^{-1}\hat{x}\right) = f_p(\hat{x}). \tag{17}$$

The resulting function also satisfies Condition 1 with the strong convexity constant ρ_p and Lipschitz constants L_p.

Let $V \in R^{n \cdot n}$ be a non-singular matrix such that for the strong convexity and Lipschitz constants of functions $f_V(\hat{x})$ with

$$\hat{x} = Vx \tag{18}$$

and $f_p(\hat{x}),$ for an arbitrary non $-$ singular matrix $P \in R^{n \cdot n}$, the inequality takes place:

$$\frac{\rho_V}{L_V} \geq \frac{\rho_p}{L_p} \tag{19}$$

Transformation (18) subsequently plays the role of a selected coordinate system, the best in terms of the convergence rate of gradient methods. Due to the fact that the gradient method, unlike Newton's method, is not invariant under a linear coordinate transformation, we cannot use the strong convexity and Lipschitz constants in the preferred coordinate system (18).

Theorem 3. *Let the function be twice differentiable and satisfy Condition 1. Then, for the sequence of iterations j = 0, 1,. . ., k process (5), (6) with the choice of the Newton's method direction (13), the following estimate holds:*

$$f_{k+1} - f^* \leq Q_k(f_0 - f^*), \quad Q_k = \prod_{j=0}^{k} q_j, \quad q_j = 1 - \frac{\rho_V^2}{L_V^2}, \; j = 0, 1, \ldots, k, \tag{20}$$

corresponding to the selected coordinate system (18), which has property (19).

Proof of Theorem 3. The iteration of Newton's method (5), (6), (13) with exact one-dimensional descent (6), has the form:

$$x_{k+1} = x_k - \beta_k \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k). \tag{21}$$

Characteristics of functions $f(x)$ and $f_p(\hat{x})$, taking into account (16) and (17), are related by:

$$f_p(\hat{x}) = f(x), \quad \nabla \hat{f}_p(\hat{x}) = \mathrm{P}^{-\mathrm{T}} \nabla f(x), \quad \nabla^2 \hat{f}_p(\hat{x}) = P^{-T} \nabla^2 f(x) P^{-1}. \tag{22}$$

After transferring process (21) to a new coordinate system, we obtain its coincidence with the method in the new coordinate system:

$$\begin{gathered} Px_{k+1} = Px_k - \beta_k P \left[\nabla^2 f(x_k)\right]^{-1} P^T P^{-T} \nabla f(x_k) = \hat{x}_k - \\ \beta_k \left[P^{-T} \nabla^2 f(x_k) P^{-1}\right]^{-1} P^{-T} \nabla f(x_k) = \hat{x}_k - \beta_k \left[\nabla^2 \hat{f}_p(\hat{x}_k)\right]^{-1} \nabla \hat{f}_p(\hat{x}_k). \end{gathered} \tag{23}$$

In the case of the relation of the initial points $\hat{x}_0 = Px_0$ for Newton's method in different coordinate systems, according to (23), at $\beta_k = \hat{\beta}_k$ sequences of points related by the $\hat{x}_k = Px_k$ and equal values of the functions $f_p(\hat{x}_k) = f(x_k)$ are generated. Moreover, taking into account the fact that the method with exact one-dimensional minimization (6) is considered, due to the extremum condition:

$$\langle s_k, \nabla f(x_k) \rangle = \left\langle P^{-1} P s_k, \nabla f(x_{k+1}) \right\rangle = \left\langle P s_k, P^{-T} \nabla f(x_{k+1}) \right\rangle = \left\langle \hat{s}_k, \nabla \hat{f}_p(\hat{x}_{k+1}) \right\rangle = 0$$

equality $\beta_k = \hat{\beta}_k$ will hold. Due to the invariance of Newton's method with respect to the linear transformation of coordinates (16), when the initial conditions $\hat{x}_0 = Px_0$ are related,

Newton's method generates identical sequences of function values in different coordinate systems. Applying the estimate in the coordinate system $\hat{x} = Vx$, taking into account the results of Theorem 2, we obtain estimate (20). □

The last estimate according to (19) determines the advantages of Newton's method compared to the gradient method in the case of:

$$\frac{\rho_V^2}{L_V^2} \gg \frac{\rho}{L}. \tag{24}$$

Taking into account the fact that when solving practical problems, most of the iterations of the method often occur under conditions of significant Hessian variation (15), estimate (20), subject to condition (24), explains the advantages of Newton's method. In this case, no additional restrictions on the second derivatives under smoothness conditions are required.

3. Subgradient Minimization Method

Here, we will give an exposition of the subgradient method [27], which solves the problem of forming the descent direction, which makes it possible to obtain a new point of the current minimum approximation by means of one-dimensional minimization along it outside a certain neighborhood of the current minimum. In this case, the appropriate direction is a vector consistent with all subgradients at points in a certain neighborhood of the current minimum approximation. In the case of smooth functions, the descent direction is matched with a set of neighborhood gradients obtained at iterations of the method.

In relaxation processes of the ε-subgradient type, successive approximations are constructed according to the formulas [39–41,43,45]:

$$x_{k+1} = x_k - \gamma_k s_{k+1},\ \gamma_k = arg\min_{\gamma} f(x_k - \gamma s_{k+1}) \tag{25}$$

The descent direction s_{k+1} is selected from a set $S(\partial_\varepsilon f(x_k))$, where $\partial_\varepsilon f(x_k)$ is ε-subgradient set at a point x_k and $S(G) = \{s \in R^n | \min_{g \in G}(s, g) > 0\}$, $G \subset R^n$ is a set of feasible directions. Denote a subgradient set at a point x by $\partial f(x) \equiv \partial f_{\varepsilon=0}(x)$. If the set $S(G)$ is not empty, then, according to its definition, any vector $s \in S(G)$ is a solution to the set of inequalities:

$$\langle s, g \rangle > 0,\ \forall g \in G, \tag{26}$$

that is, it specifies the normal of the separating plane of the origin and the set G. One of the solutions to (26) is a vector $\eta(G)$ of minimal length from G. For example, in the ε-steepest descent method, $s_{k+1} = \eta(\partial_\varepsilon f(x_k))$ [41]. Due to the absence of an explicit definition of the ε-subgradient set, in (25), the vector s that satisfies condition (26) is used as the descent direction, and the set G here is the shell of subgradients obtained on the descent trajectory [39–41].

The elements of the set G on smooth functions are the gradients of the current minimum neighborhood. Figure 1 shows the set G with the designations of its elements, which will be given below.

Denote by η_G a vector of minimum length from the set G, $\rho_G = \|\eta_G\|$, $\mu_G = \eta_G / \|\eta_G\|$, $s^* = \mu_G / \rho_G$, $R_G = \max_{g \in G} \|g\|$, $R_s = \max_{g \in G}(\mu_G, g)$, $M_G = R_S / \rho_G$. For a certain set G, we will also use the noted characteristics indicating the set as an argument, for example, $\eta(G)$, $r(G)$.

We will assume that the following assumption holds for the set G.

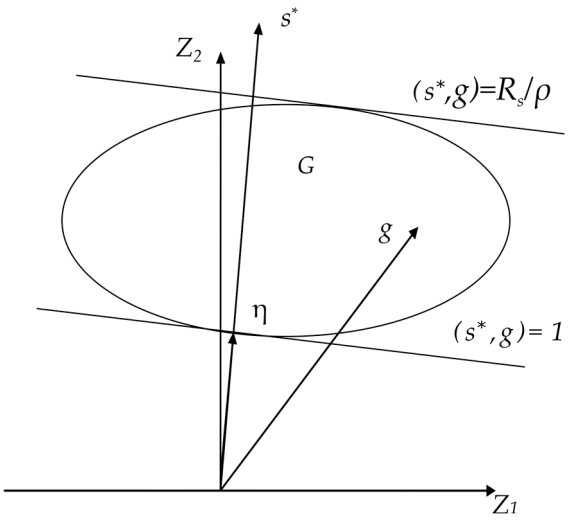

Figure 1. The set G and its characteristics [28].

Assumption 1. *Set G is convex, closed, limited ($R_G < \infty$), and satisfies the separability condition, i.e., $\rho_G > 0$.*

Let us introduce the relation $\theta(M)$ and its inverse function $m(\theta)$.

$$\theta(M) = (M-1)^2/(M+1)^2, \; m(\theta) = \left(1+\theta^{\frac{1}{2}}\right)/\left(1-\theta^{\frac{1}{2}}\right), \tag{27}$$

Thus, $m(\theta(M)) = M$. For some limited θ, define the relations:

$$a(\theta) = 1/(2\theta), \;\; b(\theta) = 1/(2(1-\theta)), \;\; 0 < \theta < 1/2. \tag{28}$$

Vector s^* is a solution to the system of inequalities (26). Parameters ρ and R_S characterize the thickness of the set G in the direction μ. The quantity R_S, according to its definition, determines the thickness of the set G and significantly affects the convergence rate of learning algorithms with space dilation. When the thickness of the set is zero, when $R_s = \rho$, we have the case of a flat set.

The quantity M_G determines the complexity of solving system (26). The transformation parameters of the metric matrices of the subgradient method are found according to expression (28).

In this work, two versions of the subgradient method will be presented. The first of them involves an exact one-dimensional search. For this version of the algorithm, estimates of the convergence rate on smooth functions will be obtained. The second version of the minimization algorithm is intended for practical implementation, where a rough one-dimensional search is used. To correctly integrate a method for solving systems of inequalities into minimization algorithms and comply with the restrictions imposed on its actions, we need to outline it. In the subgradient methods under study, the following Algorithm 1 for solving the system of inequalities (26) is used to estimate the separating plane parameters.

Algorithm 1 [27]. Algorithm for solving a system of inequalities

1. Assume $k = 0$, $H_0 = I$, $q \geq 1$. Set θ_A such that:

$$\theta(M_G) \leq \theta_A < 1/2 \tag{29}$$

and $M_A \equiv m(\theta_A)$.
2. Set $g_k \in G$ and $s_k = H_k g_k$, which is the current approximation of the solution to the system of inequalities $\langle s, g \rangle > 0 \; \forall g \in G$. Find a vector $u_k \in G$ such that:

$$\langle H_k g_k, u_k \rangle \leq 0. \tag{30}$$

If such a vector does not exist, then the solution $s_k = H_k g_k$ is found; stop the algorithm.
3. Compute vectors:

$$y_k = g_k - u_k, \quad p_k = g_k + t_k y_k, \text{ where } t_k = -\frac{\langle y_k, H_k g_k \rangle}{\langle y_k, H_k y_k \rangle}. \tag{31}$$

Here, the vector p_k, is found from the condition of vectors $v_k = H_k p_k$ and y_k orthogonality:

$$\langle y_k, H_k p_k \rangle = \langle y_k, v_k \rangle = 0. \tag{32}$$

Compute $\theta_{gk}(M_A)$, where:

$$\theta_{gk}(M) = \left(1 + \frac{\langle y_k, H y_k \rangle}{(M-1)^2 \langle p_k, H p_k \rangle} \left(1 + \frac{C_k}{\langle y_k, H y_k \rangle}(M-1)\right)^2\right)^{-1}, \tag{33}$$

$$C_k = \min\{|\langle y_k, H_k g_k \rangle|, |\langle y_k, H_k u_k \rangle|\}.$$

Find the parameter:

$$\theta_k = \begin{cases} \theta_A / q^2, & if \; \theta_{gk}(M_A) \leq \theta_A / q^2, \\ \theta_A, & if \; \theta_{gk}(M_A) \geq \theta_A, \\ \theta_{gk}(M_A), & otherwise. \end{cases} \tag{34}$$

Find the parameters according to (28):

$$\alpha_k^2 = a(\theta_k), \; \beta_k^2 = b(\theta_k). \tag{35}$$

We obtain a new approximation of the metric matrix $H_{k+1} = (H_k, \; \alpha_k, \beta_k, \; y_k, p_k)$, where:

$$(H, \; \alpha, \beta, \; y, p) = H - \left(1 - \frac{1}{\alpha^2}\right) \frac{H y y^T H^T}{\langle y, H y \rangle} - \left(1 - \frac{1}{\beta^2}\right) \frac{H p p^T H^T}{\langle p, H p \rangle}. \tag{36}$$

4. Assign $k = k + 1$. Go to step 2.

Constraint (29) for the set G in the case of applying Algorithm 1 in the minimization method imposes restrictions on the subgradient sets of the non-smooth minimization problem. In the case of smooth minimization problems, one can arbitrarily choose the parameter satisfying (29). This parameter is selected experimentally in order to optimize the algorithm efficiency.

Denote $\alpha_A^2 = a(\theta_A), \; \beta_A^2 = b(\theta_A)$. It was proven in [27] that Algorithm 1 converges in a finite number of iterations on a set G, satisfying Assumption 1, and for algorithm parameters V_0 and θ_A, for which the restrictions $0 < V_0 \leq \rho_G^2 / R_G^2$ and (29) are satisfied. In this case, the number of iterations does not exceed k_0—the minimum integer number from the range of values k satisfying the inequality:

$$\frac{25k(q^2 \alpha_A^2 - 1)}{n V_0^2 [(\alpha_A^2 \beta_A^2)^{k/n} - 1]} < 1$$

From the above estimate, the conclusion can be made that larger values of $\alpha_A^2 \beta_A^2$ correspond to fewer number of iterations k_0, which means that the desired direction will be found

in fewer number of iterations. The last estimate is based on the worst-case scenario, when all $\alpha_k^2\beta_k^2 = \alpha_A^2\beta_A^2$. In fact, according to the results of a computational experiment in [27], a minimization algorithm based on Algorithm 1 with parameter (35) is more effective than with fixed parameters $\alpha_A^2\beta_A^2$.

The version of the minimization algorithm presented in this section uses exact one-dimensional descent and is intended to estimate its convergence rate on smooth functions. A practically implementable version of the algorithm without exact one-dimensional descent will be presented in the next section. Here, as well as in the practically implemented version of the algorithm, there are no updates for the parameters of the algorithm for solving systems of inequalities in the form of setting $H_k = I$, which are used in the theoretical version of the algorithm from [27], necessary for the theoretical justification of the convergence of the minimization algorithm on non-smooth functions. In the version of the minimization algorithm used in practice when minimizing both smooth and non-smooth functions, the above update is absent, but there are minor changes to the diagonal elements of the matrix $H_k \to H_k + \lambda I$, excluding its poor conditionality and scaling, and $H_k \to cH_k\, c > 1$, excluding excessive reduction of its elements. Therefore, the described version of the algorithm is closest to the implemented versions designed to minimize smooth and non-smooth functions. As before, to denote both the gradient and the subgradient at some point x_k, we will use the notation $g_k = g(x_k) \in \partial f(x_k)$.

At Step 2 of Algorithm 1, vector $g_k \in G$ is given arbitrarily, vector $u_k \in G$ having property (30) is found in the set. In the minimization algorithm, we assume $g_k \in \partial f(x_k)$ at the point of current minimum approximation, determine the descent direction $s_k = H_k g_k$, and find the new minimum approximation $x_{k+1} = x_k - \gamma_k s_{k+1}$.

In the case of exact one-dimensional minimization, the equality $\langle g_{k+1}, s_k\rangle = \langle g_{k+1}, H_k g_k\rangle$ holds for the gradient at a point x_{k+1}. Therefore, in Algorithm 1, built into the minimization algorithm, we can take the vectors $H_k g_k$ and $u_k = g_{k+1}$ as a new pair of vectors in (30), for which, due to exact one-dimensional descent, an inequality similar to (30) will be satisfied $\langle H_k g_k, g_{k+1}\rangle = 0$. Due to the arbitrary choice of vector g_k in Algorithm 1, at the next iteration in the minimization algorithm, the vector g_{k+1} can be chosen. An idealized version of such a minimization algorithm is Algorithm 2 described below. Estimation of the convergence rate of this algorithm is the goal of our work.

In the case of inexact one-dimensional descent, it is assumed that the one-dimensional minimum has been localized, that is, a point $z_{k+1} = x_k - \gamma_z s_k$ has been obtained such that the subgradient u_{k+1} at the extreme point z_{k+1} satisfies the inequality (30) $\langle H_k g_k, u_{k+1}\rangle = \langle s_k, u_{k+1}\rangle \le 0$ (Figure 2). Subgradient u_{k+1} will be used for the matrix H_k transformation. Figure 2 shows the point x_{k+1} with the smallest found function value, which, at the next iteration, will become the new current minimum point with the direction of minimization $s_{k+1} = H_{k+1} g_{k+1}$. A presentation of the practical version of the algorithm and its numerical analysis will be given in subsequent sections.

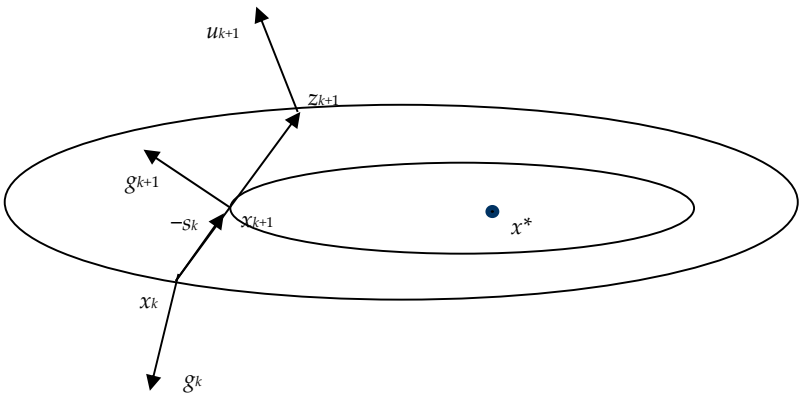

Figure 2. Selection of subgradient vectors for inexact one-dimensional descent in the method of solving systems of inequalities.

The following minimization algorithm assumes exact one-dimensional descent. An infinite sequence of points x_k is constructed until the gradient becomes zero. For this version of the algorithm, an estimate of the convergence rate has been made.

Algorithm 2. Minimization algorithm

1. Assume $k = 0$, $H_0 = I$, $q \geq 1$. Set θ_A such that:

$$\theta_A < 1/2 \tag{37}$$

and $M_A \equiv m(\theta_A)$. Compute $g_0 = \nabla f(x_0)$, If $g_0 = 0$ then the minimum point is found, stop the algorithm.
2. Find a new minimum approximation:

$$x_{k+1} = x_k - \gamma_k s_k,\ s_k = H_k g_k,\ \gamma_k = \underset{\gamma}{\operatorname{argmin}} f(x_k - \gamma s_k). \tag{38}$$

3. Compute the gradient $g_{k+1} = \nabla f(x_{k+1})$ based on the condition:

$$\langle g_{k+1} s_k \rangle \leq 0. \tag{39}$$

If $g_{k+1} = 0$, then x_{k+1} is the minimum point; stop the algorithm.
4. Compute vectors y_k, p_k:

$$y_k = g_k - g_{k+1},\ \ p_k = g_{k+1} + t_k y_k, \quad t_k = -\frac{\langle y_k, H_k g_{k+1} \rangle}{\langle y_k, H_k y_k \rangle}. \tag{40}$$

Here, the vector p_k is found from (32) based on the orthogonality of $H_k p_k$ and y_k.
Compute $\theta_{gk}(M_A)$ according to formula (33), where:

$$C_k = \min\{|\langle y_k, H_k g_k \rangle|, |\langle y_k, H_k g_{k+1} \rangle|\}. \tag{41}$$

Find θ_k according to (34) and $\alpha_k^2 = a(\theta_k)$, $\beta_k^2 = b(\theta_k)$, as in (35). We obtain a new approximation of the metric matrix $H_{k+1} = (H_k,\ \alpha_k, \beta_k,\ y_k, p_k)$,
5. Assign $k = k + 1$. Go to step 2

Here, the built-in method for solving the system of inequalities (26) is the transformations carried out at step 4 under condition (39). The solution to system (26) at iteration is the vector $H_{k+1} g_{k+1}$, which is used as the new descent direction.

In [27], the optimization of the parameters' α_k, β_k choice is related to the characteristics of the subgradient sets of the function. In [27], it is assumed that it is possible to choose a parameter M_A corresponding to the real characteristic M_ε, which is the union of subgradient sets of a certain ε-neighborhood of the current minimum point satisfies the relation:

$$\theta(M_\varepsilon) = \theta(M_A) < 1/2 \tag{42}$$

In the case of smooth functions, due to the fact that the subgradient coincides with the gradient, and the subgradient set contains a single element—the gradient—it is easy to satisfy condition (42), since for small ε, the characteristics of the subgradient set:

$$R(\partial f(x_k)) = \rho(\partial f(x_k)) = 1,\ M = M(\partial f(x_k)) = 1,$$

due to the fact that the gradient satisfies the Lipschitz condition, they change insignificantly. Therefore, for small ε:

$$M_\varepsilon = M(\partial_\varepsilon f(x_k)) = R_S(\partial_\varepsilon f(x_k))/\rho(\partial_\varepsilon f(x_k)) \approx 1,\ \theta(M_\varepsilon) = (M_\varepsilon - 1)^2/(M_\varepsilon + 1)^2 \approx 0,$$

which makes it possible to consider the algorithm for sufficiently large values of ε-neighborhoods that satisfy condition (42).

The smaller $\theta(M)is$, the more efficient the algorithm for solving systems of inequalities works [27]. But for small values of $\theta(M)$, according to (35), the values of α^2 will be very large. This will lead to large changes in the matrix H (36), which negatively affects the efficiency of the minimization method due to the difficulties that arise with the degeneration of metric matrices. Therefore, in the minimization algorithm, the smallest value θ_k has to be limited and consistent with the accuracy of the one-dimensional search. To do this, a constraint on the parameter θ_k is introduced into (34):

$$\theta(M_A)/q^2 \leq \theta_k \leq \theta(M_A) < 1/2. \tag{43}$$

As a result, we obtain restrictions on the parameters of matrix transformation in (36):

$$\alpha^2_{\max} = \frac{1}{2\theta(M_A)/q^2} \geq \alpha_k^2 \geq \frac{1}{2\theta(M_A)} = \alpha^2_{\min}, \tag{44}$$

$$\beta^2_{\min} = \frac{1}{2(1-\theta(M_A)/q^2)} \leq \beta_k^2 \leq \frac{1}{2(1-\theta(M_A))} = \beta^2_{\max}. \tag{45}$$

Relation $\theta_k(1-\theta_k)$ subject to restrictions on θ_k (43) is monotonically increasing on the segment $0 < \theta_k < 1/2$. Hence the constraints:

$$\frac{1}{4(1-\theta(M_A)/q^2)(\theta(M_A)/q^2)} \geq \alpha_k^2\beta_k^2 = \frac{1}{4\theta_k(1-\theta_k)} \geq \frac{1}{4\theta(M_A)(1-\theta(M_A))}. \tag{46}$$

From here and (44), (45) the inequalities follow:

$$\alpha^2_{\max}\beta^2_{\min} \geq \alpha_k^2\beta_k^2 \geq \alpha^2_{\min}\beta^2_{\max}. \tag{47}$$

For the parameters α_k, β_k according to (44), (45), and (46), the inequalities hold:

$$\alpha_k > 1, \quad 0 < \beta_k \leq 1, \quad \alpha_k \cdot \beta_k > 1. \tag{48}$$

The presented algorithm for solving systems of inequalities and the minimization algorithm also converge for fixed parameters $\alpha_k = \alpha_{const}, \beta_k = \beta_{const}$ [27].

$$\alpha_k^2 = \frac{1}{2\theta(M_A)} = \alpha^2_{\min}, \quad \beta_k^2 = \frac{1}{2(1-\theta(M_A))} = \beta^2_{\max}. \tag{49}$$

As a computational experiment shows, the convergence rate of the method for solving systems of inequalities and the minimization method based on it [27] is significantly higher if parameters α_k, β_k adjusted depending on the current situation (35) are used.

4. On the Convergence Rate of the Subgradient Method on Strongly Convex Functions with Lipschitz Gradient

As earlier, x^* is a minimum point of the function $f(x)$, $f^* = f(x^*)$, $f_k = f(x_k)$ and $g_k = g(x_k) = \nabla f(x_k)$ for a differentiable function satisfying Condition 1. Denote $A_k = H_k^{-1}$, $Sp(A)$ is a trace of matrix A, det A is a determinant of matrix A. For an arbitrary matrix $A > 0$, we denote $A^{1/2}$ as a symmetric matrix for which $A^{1/2} > 0$ and $A^{1/2}\ A^{1/2} = A$. For characteristics of matrices A_k, H_k we used the result from [27], valid for arbitrary parameters α_k, β_k satisfying condition (48).

Lemma 1 [27]. *Let* $H_k > 0$, *matrix* H_{k+1} *obtained as a result of transformation* $H_{k+1} = (H_k, \alpha_k, \beta_k, y_k, p_k)$, *where parameters* α_k, β_k *satisfy condition (48), and for arbitrary vectors* $y_k \neq 0$, $p_k \neq 0$, *equality (38) is satisfied. Then,* $H_{k+1} > 0$ *and:*

$$A_{k+1} = A_k + (\alpha_k^2 - 1)\frac{y_k y_k^T}{\langle y_k, H_k y_k\rangle} + (\beta_k^2 - 1)\frac{p_k p_k^T}{\langle p_k, H_k p_k\rangle}, \tag{50}$$

$$Sp(A_{k+1}) = Sp(A_k) + (\alpha_k^2 - 1)\frac{\langle y_k, y_k\rangle}{\langle y_k, H_k y_k\rangle} + (\beta_k^2 - 1)\frac{\langle p_k, p_k\rangle}{\langle p_k, H_k p_k\rangle}, \tag{51}$$

$$detH_{k+1} = \det H_k/\alpha_k^2\beta_k^2,\ detA_{k+1} = \alpha_k^2\beta_k^2 detA_k. \tag{52}$$

The following theorem shows that the presence of motion as a result of iterations (5), (6) lead to a decrease in the function.

Theorem 4. *Let the function f(x) satisfy Condition 1. Then, for the sequence* $\{f_k\}$, $k = 0, 1, 2\ldots$ *given by the process (5), (6) the following estimation takes place:*

$$f_{k+1} - f^* \le (f_0 - f^*)\exp\left[-\frac{\rho^2}{L^2}\sum_{i=0}^{k}\frac{\|y_i\|^2}{\|g_i\|^2}\right], \tag{53}$$

where $y_i = g_{i+1} - g_i$.

Proof of Theorem 4. For a strongly convex function, inequality (2) is satisfied. Taking this inequality into account, we obtain:

$$\begin{aligned} f_{k+1} - f^* = (f_k - f^*) - (f_k - f_{k+1}) = (f_k - f^*)\left(1 - \tfrac{f_k - f_{k+1}}{f_k - f^*}\right) \\ \le (f_k - f^*)\left(1 - \tfrac{2\rho(f_k - f_{k+1})}{\|g_k\|^2}\right) \end{aligned} \tag{54}$$

Inequality (3) is also valid for the one-dimensional function:

$$\phi(t) = f(x_k - ts_k/\|s_k\|).$$

From here, taking into account the exact one-dimensional search, inequality (3) and Lipschitz condition (1), the estimate follows:

$$f_k - f_{k+1} \ge \rho||x_k - x_{k+1}||^2/2 \ge \rho\frac{||y_k||^2}{2L^2}.$$

Transform (54) using the last relation and inequality $\exp(-c) \ge 1 - c,\ c \ge 0$.

$$f_{k+1} - f^* \le (f_k - f^*)\left(1 - \frac{\rho^2\|y_k\|^2}{L^2\|g_k\|^2}\right) \le (f_k - f^*)\exp\left(-\frac{\rho^2\|y_k\|^2}{L^2\|g_k\|^2}\right).$$

Recurrent use of the last inequality leads to estimate (53). □

Let us estimate the convergence rate of Algorithm 2 under more general restrictions on the parameters $\alpha_k^2\beta_k^2$.

$$a_M \ge \alpha_k^2 \ge a_m,\ \ 1 \ge \beta_k^2 \ge b_{\mathrm{m}},\ \ a_m \ge 1/b_{\mathrm{m}}. \tag{55}$$

This implies the constraint:

$$a_M \ge \alpha_k^2\beta_k^2 \ge a_m b_{\mathrm{m}}.$$

The following theorem substantiates the linear convergence rate of Algorithm 2 under constraints (55).

Theorem 5. *Let the function f(x) satisfy Condition 1. Then, for the sequence* $\{f_k\}$, $k = 0, 1, 2\ldots$ *given by the Algorithm 2 with limited initial matrix* H_0*:*

$$m_0 \le \langle H_0 z, z\rangle/\langle z, z\rangle \le M_0, \tag{56}$$

(1) with an arbitrary parameter $\alpha_k^2\beta_k^2$ satisfying (55), the following estimation takes place:

$$f_{k+1} - f^* \le (f_0 - f^*)\exp\left\{-\frac{\rho^2(k+1)}{L^2 n}\left[\frac{2\ln(a_{\mathrm{m}}b_{\mathrm{m}})}{(a_M-1)} + \frac{n\ln(m_0/M_0)}{(k+1)(a_M-1)}\right]\right\}, \quad (57)$$

(2) with parameters $\alpha_k^2\beta_k^2$ specified in Algorithm 2, the estimation is:

$$f_{k+1} - f^* \le (f_0 - f^*)\exp\left\{-\frac{\rho^2(k+1)}{L^2 n}\left[\frac{2\ln(\alpha_{\min}^2\beta_{\max}^2)}{(\alpha_{\max}^2-1)} + \frac{n\ln(m_0/M_0)}{(k+1)(\alpha_{\max}^2-1)}\right]\right\}. \quad (58)$$

Proof of Theorem 5. Based on (50), we obtain (51). Transform (51) taking into account $\beta_k^2 - 1 \le 0$, we obtain an estimate for the trace of matrices A_k:

$$Sp(A_{k+1}) \le S_P(A_k)\left(1 + \frac{(\alpha_k^2-1)\langle y_k, y_k\rangle}{Sp(A_k)\langle H_k y_k, y_k\rangle}\right) \quad (59)$$

Due to exact one-dimensional descent (38), the following condition is satisfied:

$$\langle s_k, g_{k+1}\rangle = \langle H_k g_k, g_{k+1}\rangle = 0,$$

which, together with the positive definiteness of the matrices, proves the inequality:

$$\langle H_k y_k, y_k\rangle = \langle H_k g_k, g_k\rangle + \langle H_k g_{k+1}, g_{k+1}\rangle - 2\langle H_k g_k, g_{k+1}\rangle \ge \langle H_k g_k, g_k\rangle.$$

Hence, taking into account $Sp(A_k) \ge M_k$, where M_k is the maximum eigenvalue of the matrix A_k, we obtain:

$$Sp(A_k)\langle H_k y_k, y_k\rangle \ge S_P(A_k)\langle H_k g_k, g_k\rangle \ge \frac{Sp(A_k)}{M_k}\langle g_k, g_k\rangle \ge \langle g_k, g_k\rangle.$$

Based on the last estimate, inequality (59) is transformed to the form:

$$Sp(A_{k+1}) \le S_P(A_k)\left(1 + (\alpha_k^2-1)\frac{\|y_k\|^2}{\|g(x_k)\|^2}\right). \quad (60)$$

Based on the relationship between the arithmetic mean and geometric mean of the matrix $A > 0$ eigenvalues, we have $Sp(A)/n \ge [\det(A)]^{1/n}$. From here and (60), (52) in the case of restrictions on parameters $\alpha_k^2\beta_k^2$ (55), we obtain:

$$\frac{Sp(A_0)}{n}\prod_{i=0}^{k}\left[1 + (\alpha_k^2-1)\frac{\|y_i\|^2}{\|g(x_i)\|^2}\right] \ge \frac{Sp(A_{k+1})}{n} \ge (\det(A_{k+1}))^{1/n} = \left[\prod_{i=0}^{k}\left[\alpha_i^2\beta_i^2\det(A_0)\right]\right]^{1/n} \ge \left[\left(a_m b_m\right)^{k+1}\det(A_0)\right]^{1/n}$$

and in case of choosing parameters $\alpha_k^2\beta_k^2$, as in Algorithm 2, taking into account (47), we obtain an estimate:

$$\frac{Sp(A_0)}{n}\prod_{i=0}^{k}\left[1 + (\alpha_k^2-1)\frac{\|y_i\|^2}{\|g(x_i)\|^2}\right] \ge \frac{Sp(A_{k+1})}{n} \ge (\det(A_{k+1}))^{1/n} = \left[\prod_{i=0}^{k}\left[\alpha_i^2\beta_i^2\det(A_0)\right]\right]^{1/n} \ge \left[\left(\alpha_{min}^2\beta_{max}^2\right)^{k+1}\det(A_0)\right]^{1/n}.$$

The last inequalities based on ratio $1 + p \le \exp(p)$, transform to the form:

$$\frac{Sp(A_0)}{n}exp\left[(a_M-1)\sum_{i=0}^{k}\frac{\|y_i\|^2}{\|g(x_i)\|^2}\right] \ge \left[\left(a_m b_m\right)^{(k+1)/n}\det(A_0)\right]^{1/n}, \quad (61)$$

$$\frac{Sp(A_0)}{n}exp\left[\left(\alpha_{max}^2-1\right)\sum_{i=0}^{k}\frac{\|y_i\|^2}{\|g(x_i)\|^2}\right] \ge \left[\left(\alpha_{min}^2\beta_{max}^2\right)^{(k+1)/n}det(A_0)\right]^{1/n}. \quad (62)$$

Due to condition (55) $Sp(A_0)/n \le 1/m_0$, $[\det(A)]^{\frac{1}{n}} \ge 1/M_0$. Taking logarithms of (61) and (62), taking into account the last inequalities, we find:

$$\left[(a_M - 1)\sum_{i=0}^{k}\frac{\|y_i\|^2}{\|g(x_i)\|^2}\right] \ge (k+1)\ln(a_{\mathrm{m}}b_{\mathrm{m}})/n + \ln(1/M_0) - \ln\left(\tfrac{1}{m_0}\right),$$
$$\left[(\alpha^2_{\max} - 1)\sum_{i=0}^{k}\frac{\|y_i\|^2}{\|g(x_i)\|^2}\right] \ge (k+1)\ln(\alpha^2_{\min}\beta^2_{\max})/n + \ln(1/M_0) - \ln\left(\tfrac{1}{m_0}\right),$$

This implies:

$$\sum_{i=0}^{k}\frac{\|y_i\|^2}{\|g(x_i)\|^2} \ge \frac{2(k+1)\ln(a_{\mathrm{m}}b_{\mathrm{m}})}{n(a_M-1)} + \frac{\ln(m_0/M_0)}{(a_M-1)},$$
$$\sum_{i=0}^{k}\frac{\|y_i\|^2}{\|g(x_i)\|^2} \ge \frac{2(k+1)\ln(\alpha^2_{\min}\beta^2_{\max})}{n(\alpha^2_{\max}-1)} + \frac{\ln(m_0/M_0)}{(\alpha^2_{\max}-1)},$$

which, together with estimate (53) of Theorem 4, proves (57) and (58). □

Estimating the convergence rate of Algorithm 2 under more general constraints (55) on parameters $\alpha_k^2\beta_k^2$ makes it possible to use parameters different from those generated in Algorithm 2. The paper presents a computational experiment where the parameters of Algorithm 2 were changed as follows:

$$\alpha_k^2 \to \alpha_k^2 \times \beta_k^2/c\ ,\ \beta_k^2 \to c\,. \tag{63}$$

Here, parameters c were set as follows: c = {0.2; 0.1; 0.05}. As a result of a computational experiment, it was revealed that in ill-conditioned problems such changes increase the efficiency of the minimization method, including in non-smooth optimization problems. For non-smooth problems, there is no theoretical justification for convergence under transformation (63).

The obtained estimates do not explain the fact of the high convergence rate the method, for example, on quadratic functions. To justify the accelerating properties of the method, we need to show its invariance with respect to the linear transformation of coordinates and then use estimate (58) in the coordinate system with maximal ratio ρ/L. A similar possibility exists, for example, in the case of quadratic functions, where this ratio will be equal to 1.

Let us establish a relation between the characteristics of Algorithm 2, used to minimize the functions $f(x)$ and $f_p(\hat{x})$ from (17).

Theorem 6. *Let the initial conditions of Algorithm 2, used to minimize the functions* $f(x)$ and $f_p(\hat{x})$, defined in (17), be related by the equalities:

$$\hat{x}_0 = Px_0, \qquad \hat{H}_0 = PH_0P^T. \tag{64}$$

Then, the characteristics of these processes are related by the relations:

$$f_P(\hat{x}_k) = f(x_k),\ \hat{x}_k = Px_k,\ \nabla f_P(\hat{x}_k) = P^{-T}\nabla f(x_k),\ \hat{H}_k = PH_kP^T,\ k = 0,1,2,\ldots \tag{65}$$

Proof of Theorem 6. For derivatives of functions $f(x)$ and $f_p(\hat{x})$, relation $\nabla f_p(\hat{x}) = P^{-T}\nabla f(x)$ holds. From this and assumption (64) follows (65) for $k = 0$. Let us assume that equalities (65) are satisfied for all k = 0,1,...,i. Let us show their feasibility for k = i + 1. From (38) with $k = i$ after multiplication by P on the left, taking into account the proven equalities (65), we obtain:

$$Px_{i+1} = Px_i - \gamma_i PH_iP^TP^{-T}\nabla f(x_i) = \bar{x}_i - \gamma_i\hat{H}_i\nabla f_P(\hat{x}_i). \tag{66}$$

Hence, according to the definition of the function f_p, at the stage of one-dimensional minimization (38), the equality $\gamma_i = \overline{\gamma}_i$ is satisfied. Therefore, the right side of (66) is the implementation of step (38) in the new coordinate system. Hence:

$$\hat{x}_i = Px_i, \;\; \nabla f_P(\hat{x}_i) = P^{-T}\nabla f(x_i), \quad \hat{y}_i = \nabla f_P(\hat{x}_{i+1}) - \nabla f_P(\hat{x}_i) = P^{-T}y_i. \tag{67}$$

Multiplying (36) with the current indices on the left by P, and on the right by P^T, taking into account (67), we obtain:

$$\begin{aligned} PH_{i+1}P^T &= PH_iP^T - \left(1 - \tfrac{1}{\alpha_i{}^2}\right)\frac{PH_iP^TP^{-T}y_iy_i{}^TP^{-1}PH^T{}_iP^T}{\left\langle y_i, P^{-1}PH_iP^TP^{-T}y_i\right\rangle} - \left(1 - \tfrac{1}{\beta_i{}^2}\right)\frac{PH_iP^TP^{-T}p_ip_i{}^TP^{-1}PH^T{}_iP^T}{\left\langle p_i, P^{-1}PH_kP^TP^{-T}p_i\right\rangle} \\ &= \overline{H}_i - \left(1 - \tfrac{1}{\alpha_i{}^2}\right)\frac{\hat{H}_i\hat{y}_i\hat{y}_i{}^T\hat{H}_i^T}{\left\langle \hat{H}_i\hat{y}_i, \hat{y}_i\right\rangle} - \left(1 - \tfrac{1}{\beta_i{}^2}\right)\frac{\hat{H}_i\hat{p}_i\hat{p}_i{}^T\hat{H}_i^T}{\left\langle \hat{H}_i\hat{p}_i, \hat{p}_i\right\rangle}, \end{aligned}$$

where the right side is the implementation of formula (36) in the new coordinate system. The denominators of the last formula establish a relationship:

$$\left\langle y_i, P^{-1}PH_iP^TP^{-T}y_i\right\rangle = \left\langle \hat{H}_i\hat{y}_i, \hat{y}_i\right\rangle, \quad \left\langle p_i, P^{-1}PH_kP^TP^{-T}p_i\right\rangle = \left\langle \hat{H}_i\hat{p}_i, \hat{p}_i\right\rangle.$$

Using the last equalities and formulas (41), (33) of Algorithm 2, we obtain:

$$\alpha_i^2 = \hat{\alpha}_i^2, \;\; \beta_i^2 = \hat{\beta}_i^2.$$

Finally, we obtain $PH_{i+1}P^T = \hat{H}_{i+1}$. Consequently, equalities (65) will also be valid for $k = i + 1$. Continuing the induction process, we obtain the proof of Theorem 6. □

For function $f_p(\hat{x})$ denote strong convexity constant by ρ_p, Lipschitz constant by L_p. Introduce the function $K(P) = \rho_p/L_p$. Denote by V the coordinate transformation matrix such that $K(V) \geq K(P)$ for an arbitrary non-singular matrices P.

Theorem 7. *Let the function f(x) satisfy Condition 1. Then, for the sequence $\{f_k\}$, k = 0,1,2... given by the Algorithm 2 with limited initial matrix H_0 according to (56)*

(1) with an arbitrary parameter $\alpha_k^2\beta_k^2$ satisfying (55), the following estimation takes place:

$$f_{k+1} - f^* \leq (f_0 - f^*)\exp\left\{-\frac{\rho_V^2(k+1)}{L_V^2 n}\left[\frac{2\ln(a_{\mathrm{m}}b_{\mathrm{m}})}{(a_M - 1)} + \frac{n\ln(m_0/M_0)}{(k+1)(a_M - 1)}\right]\right\}, \tag{68}$$

(2) with parameters $\alpha_k^2\beta_k^2$ specified in Algorithm 2, the estimation is:

$$f_{k+1} - f^* \leq (f_0 - f^*)\exp\left\{-\frac{\rho_V^2(k+1)}{L_V^2 n}\left[\frac{2\ln(\alpha_{\min}^2\beta_{\max}^2)}{(\alpha_{\max}^2 - 1)} + \frac{n\ln(m_0/M_0)}{(k+1)(\alpha_{\max}^2 - 1)}\right]\right\}. \tag{69}$$

where m_0, M_0 are the minimum and maximum eigenvalues of the matrix $\hat{H}_0 = VH_0V^T$ in the selected coordinate system (18) having property (19).

Proof of Theorem 7. According to the results of Theorem 6, we can choose an arbitrary coordinate system to estimate the convergence rate of the minimization process of Algorithm 2. Therefore, we use estimates (57) and (58) in a coordinate system with the matrix $P = V$ and obtain estimates (68) and (69). □

The first term in square brackets characterizes the constant in estimating the convergence rate of the method, and the second term characterizes the costs of setting up the metric matrix.

For the steepest descent method (scheme (5), (6) with (11)) on functions satisfying Condition 1, the order of the convergence rate is determined by expression (12) $q_k =$

$1-\rho/L$. Given that $l_V^2/L_V^2 \gg l/L$ estimate for Newton's method (20) is $q_k = 1-\rho_V^2/L_V^2$, for quasi-Newton method [27] is:

$$q_k = 1-\rho_V^3/(2L_V^3) \tag{70}$$

and estimates (68) and (69) for the subgradient method turn out to be preferable to (12). This situation arises, for example, when minimizing quadratic functions whose Hessians have a large spread of eigenvalues.

Thus, Algorithm 2 on strongly convex functions, without assuming the existence of second derivatives, has accelerating properties compared to the steepest descent method.

For sufficiently small values of the l_V^2/L_V^2 ratio, the average convergence rate of the subgradient method is given below:

$$\bar{q}_k \approx 1-\frac{\rho_V^2}{nL_V^2}\times\left[\frac{2\ln(\alpha_{\min}^2\beta_{\max}^2)}{(\alpha_{\max}^2-1)}+\frac{n\ln(m/M)}{(k+1)(\alpha_{\max}^2-1)}\right]\approx 1-\frac{2\rho_V^2}{nL_V^2}\times\frac{\ln(\alpha_{\min}^2\beta_{\max}^2)}{(\alpha_{\max}^2-1)}. \tag{71}$$

The second term in square brackets of estimate (71) characterizes the stage of adjusting the metric matrix of Algorithm 2. From the analysis of expression (71), we can conclude that the qualitative nature of estimate (69) is similar to estimate (20) for Newton's method, which takes into account the difference between information in the form of a matrix of second derivatives in (20) and a gradient in (69) through the presence of the factor $1/n$ in (69).

To test the effectiveness of the algorithm, it makes sense to implement Algorithm 2 and conduct numerical testing in order to identify its application possibilities in solving problems of minimizing smooth functions along with effective quasi-Newton methods for solving minimization problems with a high degree of conditionality.

5. Aspects of the Subgradient Method Implementation

In the case of inexact one-dimensional descent, in operation (25) of the minimization algorithm, it is assumed that the one-dimensional minimum has been localized, that is, a point $z_{k+1} = x_k - \gamma_z s_k$ has been obtained such that the subgradient u_{k+1} at the extreme point z_{k+1} satisfies inequality (30):

$$\langle H_k g_k, u_{k+!}\rangle = \langle s_k, u_{k+1}\rangle \le 0,$$

which is shown in Figure 2.

The subgradient u_{k+1} is used to transform the matrix H_k. Figure 2 shows a point x_{k+1} with a smaller function value on the localization segment between points x_k and u_{k+1}

$$f(x_k) \ge f(x_{k+1}) \le f(u_{k+1}),$$

which, at the next iteration, will become the new current minimum point with the direction of minimization $s_{k+1} = H_{k+1}g_{k+1}$.

In Algorithm 1, at each iteration vector, $g_k \in G$ is chosen arbitrarily, and then, vector $u_k \in G$ such that $\langle H_k g_k, u_k\rangle \le 0$. In the minimization algorithm with one-dimensional minimization, from a point x along the direction $s = Hg$, when carrying out localization of the minimum, we obtain a point $x_1 = x - \gamma_1 s$ for which a similar (30) inequality is satisfied, and a point $x_m = x - \gamma_m s$ inside the localization segment with a smaller function value, which we take in the minimization algorithm as a new minimum approximation from which a new one-dimensional descent will then be carried out. Gradients $g_x = g(x)$ and $g_1 = g(x_1)$ at the points x and x_1 are used together for matrix transformation. Thus, in the practical version of the minimization algorithm, vectors $g_x, g_1, g(x_m)$ will be used, corresponding in meaning to the vectors $g_k \in G, u_k \in G,\ g_{k+1} \in G$ from Algorithm 1.

We use the one-dimensional minimization procedure based on these principles, outlined in [27,43]. Its set of input parameters is $\{x, s, g_x, f_x, h_0\}$, where x is the point of the current minimum approximation, s is the descent direction, h_0 is the initial search step, $f_x = f(x)$,

$g_x \in \partial f(x)$, and the necessary condition for the possibility of reducing the function along the direction $\langle g_x, s\rangle > 0$ must be satisfied. Its output parameters are $\{\gamma_m, f_m, g_m, \gamma_1, g_1, h_1\}$. Here, γ_m is the step to the point of a new minimum approximation:

$$x_m = x - \gamma_m s, \quad f_m = f(x_m), \quad g_m \in \partial f(x_m),$$

γ_1 is the step along s such that at the point $x_1 = x - \gamma_1 s$ for the subgradient $g_1 \in \partial f(x_1)$ inequality $\langle g_1, s\rangle \le 0$ holds. This subgradient is used in the learning algorithm. The output parameter h_1 is the initial descent step for the next iteration. The step h_1 is adjusted to reduce the number of calls to the procedure for calculating the function and subgradient.

In the minimization algorithm, the vector $g_1 \in \partial f(x_1)$ is used to solve a system of inequalities, and the point $x_m = x - \gamma_m s$ as the point of a new minimum approximation.

We denote the call to the procedure as OM($\{x, s, g_x, f_x, h_0\}$; $\{\gamma_m, f_m, g_m, g_1, h_1\}$). Here is a brief description of it.

Let us introduce a one-dimensional function $\varphi(\beta) = f(x - \beta s)$. To localize its minimum, we take an increasing sequence $\beta_0 = 0$, $\beta_i = h_0 q_M^{i-1}$, $i \ge 1$. Here, $q_M > 1$ is a step increasing parameter. In most cases, it is specified $q_M = 3$. Denote $z_i = x - \beta_i s$, $r \in \partial f(z_i)$, $i = 0, 1, 2, \ldots$, l is number of i at which the relation $\langle r_i, s\rangle \le 0$ is first time satisfied. Let us determine the parameters of the localization segment $[\gamma_0, \gamma_1]$ of one-dimensional minimum: $\gamma_0 = \beta_{l-1}$, $f_0 = f(z_{l-1})$, $g_0 = r_{l-1}$, $\gamma_1 = \beta_l$, $f_1 = f(z_l)$, $g_1 = r_{l-1}$ and find a minimum point γ^* through cubic approximation of the function [46] on the localization segment, using the values of the one-dimensional function and its derivative. Calculate:

$$\gamma_m = \begin{cases} 0.1\gamma_1, & if\ \ l = 1\ \ and\ \ \gamma^* \le 0.1\gamma_1, \\ \gamma_1, & if\ \ \gamma_1 - \gamma^* \le 0.2(\gamma_1 - \gamma_0), \\ \gamma_0, & if\ \ l > 1\ \ and\ \ \gamma^* - \gamma_0 \le 0.2(\gamma_1 - \gamma_0), \\ \gamma^*, & otherwise. \end{cases}$$

We calculate the initial descent step for the next iteration using the rule:

$$h_1 = q_m h_0 (\gamma_1 / h_0)^{1/2}.$$

Here, $q_m < 1$ is descent step decreasing parameter, which, in most cases, is set as $q_m = 0.8$. In the vast majority of applications, the set of parameters $\{q_M = 3, q_m = 0.8\}$ is satisfactory. When solving complex problems with a high degree of level surfaces elongation, the parameter should be increased: $q_m \to 1$. Subgradient method implementation is presented in Algorithm 3.

Here, the built-in method for solving the system of inequalities (26) is the transformations carried out at Step 3 under condition (39). The current approximation of the solution to system (26) at the iteration is vector s_k (74), which is used as the new descent direction.

The algorithm uses soft matrix updating due to small changes in diagonal elements in the case of large angles (72) between vectors s_k and g_k. Due to the fact that as a result of matrix transformations, its elements are reduced to compensate for this effect, a scaling transformation (75) is carried out, which does not affect the computational process. Taking into account the scaling of the descent direction (74), simultaneously with the scaling of the matrix, the one-dimensional search step is also scaled, which is adjusted in the one-dimensional minimization procedure.

Along with formula (33), we used a simplified version of calculating the value of $\theta_{gk}(M_A)$, which enables us to analyze the qualitative nature of formula (33). Using symmetric matrix $H_k^{1/2}$, we form vectors $a = H_k^{1/2} y_k$, $b = H_k^{1/2} g_{k+1}$, $c = H_k^{1/2} g_k$, $p = H_k^{1/2} p_k$ and assume equality $\|a\| = \|b\|$. Hence, due to the equality $a = b - c$, the vectors a, b, c form an isosceles triangle (Figure 3).

Algorithm 3. Subgradient method implementation

1. Assume $k = 0$, initial matrix $H_0 = I$, $q \geq 1$, the number of iterations k_{max} to stop the algorithm. Set Θ_A satisfying inequality (37), and parameter $M_A \equiv m(\theta_A)$. Compute $g_0 \in \partial f(x_0)$. Set the initial step of a one-dimensional search h_0 and small $\varepsilon = 10^{-10}$. If $g_0 = 0$ and then the x_0 is a minimum point, stop the algorithm.
2. If

$$\frac{\langle s_k, g_k \rangle}{\|s_k\| \times \|g_k\|} \leq \varepsilon, \tag{72}$$

then correct the matrix:

$$H_k = H_k + 10\varepsilon d_{max} I,\ d_{max} = \max_i \{H_{ii,k}\},\ i = 1, 2, \ldots, n. \tag{73}$$

Set

$$s_k = H_k g_k / \langle H_k g_k, g_k \rangle^{1/2}. \tag{74}$$

Find a new minimum approximation:

$$\mathrm{OM}(\{x_k, s_k, g_k, f_k, h_k\};\ \{\gamma_{k+1}, f_{k+1}, g_{k+1}, u_{k+1}, h_{k+1}\}).$$

According to the description of the OM procedure, here the subgradient vector u_{k+1} satisfies the condition $\langle H_k g_k, u_{k+1} \rangle \leq 0$.
If $g_{k+1} = 0$ then x_{k+1} is the minimum point, stop the algorithm.
If $k > k_{max}$ then stop the algorithm.
3. Compute vectors $y_k,\ p_k$ by (31).

$$y_k = g_k - u_{k+1}, \qquad t_k = -\frac{\langle y_k, H_k g_{k+1} \rangle}{\langle y_k, H_k y_k \rangle}, \qquad p_k = g_{k+1} + t_k y_k.$$

Here, vector p_k is found from the orthogonality condition (32) of the vectors $H_k p_k$ and y_k. Then, compute $\theta_{gk}(M_A)$ by (33), where C_k is calculated by formula (41).
Find θ_k according to (34) and parameters $\alpha_k^2 = a(\theta_k),\ \beta_k^2 = b(\theta_k)$ by (35). We obtain a new approximation of the metric matrix $H_{k+1} = (H_k,\ \alpha_k, \beta_k,\ y_k, p_k)$,
If $d_{max} \leq \varepsilon$, then carry out scaling

$$H_{k+1} = H_{k+1}/d_{\max}, \quad h_{k+1} = h_{k+1}\sqrt{d_{\max}}, \quad d_{\max} = \max_{i=1,2,\ldots,n} \{H_{ii,k+1}\} \tag{75}$$

4. Assign $k = k + 1$. Go to step 2.

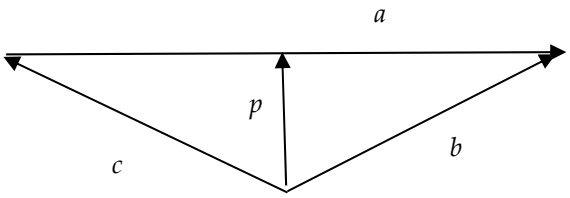

Figure 3. Properties of vectors a, b, c, p.

Due to the fact that the lengths of the vectors b, c projections onto the vector a are the same, the equality $|\langle a, b \rangle| = |\langle a, c \rangle| = \langle a, a \rangle / 2$ holds. Therefore:

$$C_k = \min\{|\langle y_k, H_k g_k \rangle|, |\langle y_k, H_k g_{k+1} \rangle|\} = |\langle y_k, H_k g_k \rangle| = |\langle a, c \rangle| = \langle a, a \rangle / 2 = \langle y_k, H_k y_k \rangle / 2$$

and the factor from (33) can be transformed as follows:

$$1 + \frac{C_k}{\langle y_k, H_k y_k \rangle}(M - 1) = 1 + \frac{\langle y_k, H_k y_k \rangle}{2\langle y_k, H_k y_k \rangle}(M - 1) = \frac{M + 1}{2}.$$

From here and (33), we obtain:

$$\theta_{gk}(M) = \left(1 + \frac{\langle y_k, Hy_k\rangle}{(M-1)^2\langle p_k, Hp_k\rangle}\left(1 + \frac{C_k}{\langle y_k, H_k y_k\rangle}(M-1)\right)^2\right)^{-1} = \left(1 + \frac{(M+1)^2\langle y_k, H_k y_k\rangle}{4(M-1)^2\langle p_k, H_k p_k\rangle}\right)\right)^{-1}$$
$$\approx \left(\frac{(M+1)^2\langle y_k, H_k y_k\rangle}{4(M-1)^2\langle p_k, H_k p_k\rangle}\right)\right)^{-1} = \frac{(M-1)^2}{(M+1)^2} \times \frac{4\langle p_k, H_k p_k\rangle}{\langle y_k, H_k y_k\rangle} = \theta(M) \times \frac{4\langle p_k, H_k p_k\rangle}{\langle y_k, H_k y_k\rangle} = \theta(M) \times \frac{4\langle p, p\rangle}{\langle a, a\rangle} \quad (76)$$

At the last steps of the transformation in (76), we used the expression $\theta(M) = (M-1)^2/(M+1)^2$ introduced earlier in (27). As shown in [27], Algorithm 2 is also operable when using formula (27) $\theta(M_A) = \theta_A$ to calculate the transformation coefficients of matrices (49) instead of $\theta_{gk}(M_A)$.

Approximate formula (76) reflects the qualitative nature of the relation $\theta_{gk}(M_A)$. According to Figure 3, larger angles between vectors b, c correspond to smaller values of the ratio $\langle p, p\rangle / \langle a, a\rangle$, which, according to (76), reduces the value of $\theta_{gk}(M_A)$ and, accordingly, leads to an increase in the parameter α_k^2 and an insignificant decrease in the parameter β_k^2 at Step 3 of Algorithm 3. We used a simplified expression for $\theta_{gk}(M)$ from (76) in Algorithm 3.

Below, we present examples of solving test problems using the quasi-Newton BFGS method Algorithms 2 and 3.

6. Results of Numerical Study on Smooth Functions

Algorithms 2 and 3 were implemented with parameters $\theta_A = 0.04356$ and $M_A = 1.52755$, providing the following product: $\alpha^2\beta^2 = 1/(4\theta_A(1-\theta_A)) = 6$. These values were used in Algorithms 2 and 3 with dynamic parameters ${\alpha_k}^2{\beta_k}^2$ selection method. The methods used the one-dimensional search described above. For comparison, the quasi-Newtonian BFGS method was implemented with a one-dimensional search procedure using cubic interpolation [46]. In all methods, the function and gradient were calculated simultaneously.

Tables 1–5 show the number of calculations of function and gradient values required to achieve the designated accuracy by the function $f(x_k) - f^* \le \varepsilon$. The initial point of minimization x_0 and the value ε are given in the description of the function.

The purpose of testing is to experimentally study the ability of subgradient method and quasi-Newton method to eliminate the background that slows down the convergence rate, which is eliminated through some linear transformation that normalizes the elongation of function level surfaces in different directions, which is predicted theoretically by the estimate (69) of Theorem 7.

Due to the fact that the use of subgradient methods with changing the space metric and quasi-Newtonian methods is justified primarily on functions with a high degree of conditionality, where conjugate gradient methods do not work, the test functions were selected based on this position. Due to the fact that the quasi-Newton method is based on a quadratic model of a function, its local convergence rate in a certain neighborhood of the current minimum is largely determined by how effective it is in minimizing ill-conditioned quadratic functions. Therefore, research was primarily carried out on quadratic functions and functions of their derivatives.

If the function is twice differentiable, then the eigenvalues of the Hessian are limited by the interval of the strong convexity parameter and Lipschitz parameter [ρ,L]. Previously, we did not use second derivatives in our proofs. Nevertheless, when developing tests, we used the representation of a quadratic function and the analysis of its conditionality, relying on its eigenvalues. The test functions simulate the oscillatory nature of the second derivatives in two ways. The first of them is the drift of the corresponding eigenvalue from one value to another. In the second method, we imposed noise on the length of the gradient vector randomly, which is reflected in the calculations of the gradient difference in subgradient Algorithms 2 and 3 (40) and in the quasi-Newton method:

$$y_k = \nabla f(x_{k+1}) - \nabla f(x_k).$$

With the described methods of simulating oscillations of the Hessian imposed on some basic quadratic function with given characteristics of the eigenvalues, we have characteristics of the degeneracy degree of the problem and can set, on the one hand, the scaling that the methods under study should exclude and the degree of oscillations of the scales simulating the change matrices of second derivatives within specified limits.

The following is accepted as the basic quadratic function:

$$f_1(x, [a\max]) = \frac{1}{2}\sum_{i=1}^{n} a_i\, x_i^2,\ a_i = a\max^{\frac{i-1}{n-1}}$$

The eigenvalues a_i of this function have the limits $\lambda_{min} = 1,\ \lambda_{max} = a_{max}$. In this case, the methods under study have to remove the basic (trend) scaling specified by the coefficients of this function. To simulate random fluctuations of second derivatives, a function f_2 was created. To calculate the function values, the basic function $f_2 = f_1(x, [a_{max}])$ was used. Its gradients were distorted randomly according to the following scheme:

$$\nabla f_2 = \nabla f_1 \times (1 + r \times \xi)$$

where $\xi \in [-1,1]$ is a random number uniformly distributed on a segment $[-1,1]$, $r = 0.3$. Such function will be noted as $f_2(x, [a_{max},\ r = 0.3])$.

Here, the parameters are the base function parameters and the gradient distortion parameter. It should be noted that distortion of gradients significantly reduces the accuracy of one-dimensional search, where gradients are used to estimate directional derivatives in cubic approximation.

In the third function, additional variables c_i were used to change the scales of a_i for each of the variables.

$$c_i = \frac{b_{max}}{b_i}\left(\frac{x_i^2}{1 + x_i^2}\right) + b_i\left(1 - \frac{x_i^2}{1 + x_i^2}\right),\ b_i = b_{max}{}^{\frac{i-1}{n-1}}.$$

This function near the extremum will have the form:

$$f_3(x, [a_{max},\ b_{max}]) \approx \frac{1}{2}\sum_{i=1}^{n} a_i b_i {x_i}^2.$$

Far from the extremum, we obtain a function in which the coefficients b_i are used in reverse order:

$$f_3(x, [a_{max},\ b_{max}]) \approx \frac{1}{2}\sum_{i=1}^{n} a_i \frac{b_{max}}{b_i} {x_i}^2.$$

Changes in coefficients c_i scales have the following range: ${\lambda_{min}}^c = 1,\ {\lambda_{max}}^c = b_{max}$.

The point x_0 = (100, 100, …, 100) was chosen as initial in all the above functions. Additionally, the following nonlinear functions were also used for testing and analysis.

Function f_4 has ellipsoidal level surfaces corresponding to a quadratic function.

$$f_4(x) = \left(\sum_{i=1}^{n} x_i^2 \cdot i \cdot i\right)^2,\ x_0 = (1, 1, \ldots, 1),$$

Function f_5 has a multidimensional ellipsoidal ravine. Minimization occurs when moving along this curvilinear ravine to the minimum point.

The stopping criterion was:

$$f(x^k) - f^* \le \varepsilon = 10^{-10}.$$

Minimization results are presented in Tables 1–6. Tables 1–5 show the results of minimizing the five presented functions for various dimensions. These tables allow us to analyze the effect of removing the basic background using subgradient and quasi-Newton

methods. The cells contain: *N_it*—number of iterations (one-dimensional searches along the direction); *nfg*—number of calls to the procedure for simultaneous calculation of a function and gradient.

Table 1 shows the results of minimizing the quadratic function f_1, intended for the basic scaling of variables. This function is a background that must be removed by the method's metric matrix. The *nfg* costs of subgradient methods here are approximately two times higher compared to the BFGS method.

Table 1. Function $f_1(x, [a_{max} = 10^8])$ minimization results.

n	Algorithm 3		Algorithm 2		BFGS	
	N_it	*nfg*	*N_it*	*nfg*	*N_it*	*nfg*
100	370	784	331	696	125	276
200	527	1070	538	1096	243	523
300	738	1424	746	1430	348	746
400	934	1740	944	1779	447	948
500	1122	2084	1135	2129	542	1146
600	1298	2359	1301	2474	634	1334
700	1434	2645	1454	2695	724	1525
800	1564	2842	1598	2965	811	1710
900	1698	3056	1727	3166	897	1884
1000	1821	3280	1839	3429	982	2061

Table 2 shows the results for the function f_2. Algorithms 2 and 3 show approximately the same results. The results for the BFGS method are approximately two times worse. Gradient noise has a detrimental effect on the accuracy of one-dimensional search with cubic interpolation that use function gradients. Reducing the accuracy of one-dimensional descent has a negative impact on the BFGS method.

Table 2. Function $f_2(x, [a_{max} = 10^8,\ r = 0.3])$ minimization results.

n	Algorithm 3		Algorithm 2		BFGS	
	N_it	*nfg*	*N_it*	*nfg*	*N_it*	*nfg*
100	357	771	360	783	561	1321
200	568	1198	565	1185	1083	2564
300	769	1607	761	1607	1486	3514
400	952	1995	975	2041	1827	4335
500	1132	2323	1152	2405	2222	5279
600	1306	2673	1345	2753	2587	6152
700	1470	3048	1489	3068	2802	6652
800	1599	3311	1666	3419	3167	7566
900	1733	3581	1783	3689	3543	8442
1000	1876	3866	1930	3992	3584	8577

Table 3 shows the results of function f_3 minimization. Algorithms 2 and 3 show approximately the same results. The results for the BFGS method are approximately five times worse. In this problem, as the extremum is approached, the variables are rescaled. Possibly, this is due to differences in the degree of inclusion of the relation ρ_V / L_V. For the BFGS method according to (70) it is ${\rho_V}^3 / 2{L_V}^3$, and for subgradient methods according to (71), it is ${\rho_V}^2 / 2{L_V}^2$.

Table 3. Function $f_3(x, [a_{max} = 10^8, b_{max} = 10^2])$ minimization results.

n	Algorithm 3		Algorithm 2		BFGS	
	N_it	*nfg*	*N_it*	*nfg*	*N_it*	*nfg*
100	407	900	415	911	2654	6901
200	681	1461	696	1482	4780	11,885
300	951	1950	965	1980	6373	15,385
400	1202	2415	1221	2480	7571	17,917
500	1441	2837	1458	2912	8297	19,434
600	1653	3257	1674	3294	8968	20,900
700	1864	3672	1898	3710	9572	22,214
800	2061	4016	2108	4090	9914	22,967
900	2258	4343	2288	4411	10,391	24,001
1000	2457	4686	2481	4761	10,645	24,500

Table 4 shows the results of function f_4 minimization. Algorithms 2 and 3 show approximately the same results. The absence of quadraticity of the function while maintaining the topology of the function level surfaces, equivalent to the topology of the quadratic function, affects the convergence rate of Algorithms 2 and 3 to a lesser extent than the BFGS method. The lack of quadraticity of the function here significantly affects the convergence rate of BFGS method.

Table 4. Function f_4 minimization results.

n	Algorithm 3		Algorithm 2		BFGS	
	N_it	*nfg*	*N_it*	*nfg*	*N_it*	*nfg*
100	154	267	156	295	953	2226
200	266	443	261	438	2012	4682
300	377	619	362	604	3136	7282
400	454	737	456	741	4314	10,027
500	556	889	573	929	5523	12,815
600	669	1078	672	1095	6747	15,658
700	762	1215	778	1259	7990	18,537
800	877	1400	870	1413	9243	21,430
900	968	1545	971	1558	10,541	24,455
1000	1094	1752	1089	1765	11,746	27,226

Table 5 shows the results of minimization for function f_5. Algorithm 2 is slightly better than Algorithm 3. This function also turned out to be difficult for the BFGS method. This function, like f_4, contains polynomials of the fourth degree, which, unlike subgradient methods, significantly affects the convergence rate of the BFGS method.

Table 5. Function f_5 minimization results.

n	Algorithm 3		Algorithm 2		BFGS	
	N_it	*nfg*	*N_it*	*nfg*	*N_it*	*nfg*
100	498	1116	432	989	1170	2847
200	558	1286	450	1051	1417	3396
300	609	1423	496	1196	1700	4118
400	705	1687	442	1100	1862	4465
500	686	1653	388	980	1964	4722
600	613	1499	429	1091	2081	4955
700	581	1434	433	1106	2228	5315
800	451	1176	394	1048	2180	5200
900	533	1361	430	1135	2412	5727
1000	554	1430	435	1188	2490	5957

In Table 6, the results for functions f_1–f_5 at n = 1000 are presented. The results show the effectiveness of the methods on all functions under study simultaneously. Conclusions regarding the effectiveness of the methods were made earlier.

Table 6. Functions f_1–f_5 minimization results at $n = 1000$.

n	Algorithm 3		Algorithm 2		BFGS	
	N_it	***nfg***	***N_it***	***nfg***	***N_it***	***nfg***
f_1	1821	3280	1839	3429	982	2061
f_2	1876	3822	1930	3992	3584	8577
f_3	2457	4686	2481	4761	10,645	24,500
f_4	1094	1752	1089	1765	11,746	27,226
f_5	554	1430	435	1188	2490	5957

Table 7 shows the results of Algorithm 3 on the first three functions for $n = 1000$ with changed parameters $\alpha_k^2 \to \alpha_k^2 \times \beta_k^2/c$, $\beta_k^2 \to c$ according to (71) and different values of $c = \{0.2; 0.1; 0.05\}$. It is shown here that, on ill-conditioned problems, such changes increase the efficiency of the minimization method. These examples show the possibility of setting the method parameters for a certain fixed set of optimization problems.

Table 7. Functions f_1–f_3 minimization results at $n = 1000$, Algorithm 3, changed parameters α_k^2, β_k^2, variants of parameter c.

	f_1		f_2		f_3	
	N_it	***nfg***	***N_it***	***nfg***	***N_it***	***nfg***
no changes	1821	3280	1876	3866	2457	4686
c = 0.2	1655	2941	1800	3800	2240	4215
c = 0.1	1561	2798	1796	3756	2036	3815
c = 0.05	1625	2974	1877	3905	2175	4038

Regarding the convergence rate of minimization methods, the following conclusions can be drawn:

1. For functions close in properties to quadratic (f_1), the quasi-Newton BFGS method significantly exceeds subgradient Algorithms 2 and 3 in terms of convergence rate.
2. In the case of significant interference imposed on the gradients of the function (f_2), subgradient Algorithms 2 and 3 are more effective than the BFGS method.
3. Variability of scales across variables (f_2) affects the convergence rate of subgradient methods to a lesser extent than the BFGS method.
4. The presence of polynomial degrees higher than 2 in the minimized function affects the convergence rate of subgradient methods to a lesser extent than the BFGS method.
5. A computational experiment showed the possibility of adjusting the parameters of the method in accordance with theoretical principles. Therefore, the efficiency of the method can be increased on a certain fixed set of optimization problems.
6. Based on the performed computational experiment, it can be seen that the theoretically predicted ability of subgradient methods to exclude the background that slows down the convergence rate has been confirmed by the computational experiment.

Based on the theoretical principles and experimental results, we can conclude that the presented subgradient methods complement quasi-Newton methods when solving smooth optimization problems.

7. Conclusions

The conditionality of the minimization problem determines the spread of the elongation of level surfaces in different directions, which determines the complexity of solving the problem. In minimization practice, in many cases, it turns out to be possible to reduce the elongation of level surfaces due to some linear transformation of coordinates. The paper studies the possibility of Newton's method and the subgradient method with parameter optimization by changing the space metric to eliminate the conditionality of the problem using a linear transformation.

The paper proves that under conditions of instability of the second derivatives of the function in the minimization domain, the estimate of the convergence rate of Newton's method is determined by the strong convexity parameter and Lipschitz parameter in the coordinate system where their ratio is maximum. This means the method's ability to exclude the linear background, which increases the conditionality degree of the problem. The estimate of convergence rate serves as a standard for assessing the capabilities of the subgradient method being studied.

The paper studies RSM with parameters optimization of the rank-two correction of metric matrices on smooth, strongly convex functions with a Lipschitz gradient without assumptions about the existence of second derivatives of the function. Under broad assumptions on the transformation parameters of metric matrices, an estimate of the convergence rate of the studied RSM and an estimate of its ability to exclude removable linear background are obtained. The obtained estimates turn out to be qualitatively similar to estimates for Newton's method.

A practical version of RSM and test functions have been developed that simulate the presence of a removable linear background. A computational experiment was carried out in which the quasi-Newton BFGS method and the subgradient method under study were compared on various types of smooth functions. The testing results indicate the effectiveness of the subgradient method in minimizing smooth functions with a high degree of conditionality of the problem and its ability to eliminate the linear background that worsens the convergence.

Depending on the type of function, one or another method dominates, which allows us to conclude that the subgradient method is applicable along with quasi-Newton methods when solving problems of minimizing smooth functions with a high degree of conditionality.

Author Contributions: Conceptualization, V.K. and E.T.; methodology, V.K. and E.T.; software, V.K.; validation, L.K. and E.T.; formal analysis, E.T.; investigation, E.T.; resources, L.K.; data curation, V.K.; writing—original draft preparation, E.T. and V.K.; writing—review and editing, E.T. and L.K.; visualization, V.K. and E.T.; supervision, L.K.; project administration, L.K.; funding acquisition L.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Science and Higher Education of the Russian Federation (State Contract FEFE-2023-0004).

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Jensen, T.L.; Diehl, M. An Approach for Analyzing the Global Rate of Convergence of Quasi-Newton and Truncated-Newton Methods. *J. Optim. Theory Appl.* **2017**, *172*, 206–221. [CrossRef]
2. Nesterov, Y. A method of solving a convex programming problem with convergence rate o(1/k2). *Soviet Mathematics Doklady* **1983**, *27*, 372–376.
3. Rodomanov, A.; Nesterov, Y. Rates of superlinear convergence for classical quasi-Newton methods. *Math. Program.* **2022**, *194*, 159–190. [CrossRef]
4. Rodomanov, A.; Nesterov, Y. New Results on Superlinear Convergence of Classical Quasi-Newton Methods. *J. Optim. Theory Appl.* **2021**, *188*, 744–769. [CrossRef]
5. Jin, Q.; Mokhtari, A. Non-asymptotic superlinear convergence of standard quasi-Newton methods. *Math. Program.* **2023**, *200*, 425–473. [CrossRef]
6. Davis, K.; Schulte, M.; Uekermann, B. Enhancing Quasi-Newton Acceleration for Fluid-Structure Interaction. *Math. Comput. Appl.* **2022**, *27*, 40. [CrossRef]
7. Hong, D.; Li, G.; Wei, L.; Li, D.; Li, P.; Yi, Z. A self-scaling sequential quasi-Newton method for estimating the heat transfer coefficient distribution in the air jet impingement. *Int. J. Therm. Sci.* **2023**, *185*, 108059. [CrossRef]
8. Argyros, I.; George, S. On a unified convergence analysis for Newton-type methods solving generalized equations with the Aubin property. *J. Complex.* **2024**, *81*, 101817. [CrossRef]

9. Dennis, J.E.; Schnabel, R.B. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*; SIAM: Philadelphia, PA, USA, 1996.
10. Polak, E. *Computational Methods in Optimization*; Mir: Russia, Moscow, 1974.
11. Indrapriyadarsini, S.; Mahboubi, S.; Ninomiya, H.; Kamio, T.; Asai, H. Accelerating Symmetric Rank-1 Quasi-Newton Method with Nesterov's Gradient for Training Neural Networks. *Algorithms* **2022**, *15*, 6. [CrossRef]
12. Mokhtari, A.; Eisen, M.; Ribeiro, A. An incremental quasi-Newton method with a local superlinear convergence rate. In Proceedings of the EEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 4039–4043. [CrossRef]
13. Mokhtari, A.; Eisen, M.; Ribeiro, A. IQN: An incremental quasi-Newton method with local superlinear convergence rate. *SIAM J. Optim.* **2018**, *28*, 1670–1698. [CrossRef]
14. Liu, D.C.; Nocedal, J. On the limited memory BFGS method for large scale optimization. *Math. Program.* **1989**, *45*, 503–528. [CrossRef]
15. Berahas, A.S.; Jahani, M.; Richtárik, P.; Takác, M. Quasi-Newton Methods for Machine Learning: Forget the Past, Just Sample. *Optim. Methods Softw.* **2022**, *37*, 1668–1704. [CrossRef]
16. Mokhtari, A.; Ribeiro, A. Regularized stochastic BFGS algorithm. *IEEE Trans. Signal Proc.* **2014**, *62*, 1109–1112. [CrossRef]
17. Gower, R.; Richtárik, P. Randomized quasi-Newton updates are linearly convergent matrix inversion algorithms. *SIAM J. Matrix Anal. Appl.* **2017**, *38*, 1380–1409. [CrossRef]
18. Gao, W.; Goldfarb, D. Quasi-Newton methods: Superlinear convergence without line searches for self-concordant functions. *Optim. Methods Softw.* **2019**, *34*, 194–217. [CrossRef]
19. Byrd, R.H.; Hansen, S.L.; Nocedal, J.; Singer, Y. A stochastic quasi-Newton method for large-scale optimization. *SIAM J. Optim* **2016**, *26*, 1008–1031. [CrossRef]
20. Meng, S.; Vaswani, S.; Laradji, I.; Schmidt, M.; Lacoste-Julien, S. Fast and Furious Convergence: Stochastic Second Order Methods Under Interpolation. 2019. Available online: https://arxiv.org/pdf/1910.04920.pdf (accessed on 30 March 2024).
21. Zhou, C.; Gao, W.; Goldfarb, D. Stochastic adaptive quasi-Newton methods for minimizing expected values. In Proceedings of the 34th ICML (PMLR), Sydney, Australia, 6–11 August 2017; Volume 70, pp. 4150–4159.
22. Makmuang, D.; Suppalap, S.; Wangkeeree, R. The regularized stochastic Nesterov's accelerated Quasi-Newton method with applications. *J. Comput. Appl. Math.* **2023**, *428*, 115190. [CrossRef]
23. Rodomanov, A.; Nesterov, Y. Greedy quasi-Newton methods with explicit superlinear convergence. *SIAM J. Optim.* **2021**, *31*, 785–811. [CrossRef]
24. Lin, D.; Ye, H.; Zhang, Z. Explicit Convergence Rates of Greedy and Random Quasi-Newton Methods. *J. Mach. Learn. Res.* **2022**, *23*, 1–40.
25. Polyak, B.T. *Introduction to Optimization*; Optimization Software: New York, NY, USA, 1987.
26. Karmanov, V. *Mathematical Programming*; Mir: Moscow, Russia, 1989.
27. Krutikov, V.N.; Stanimirović, P.S.; Indenko, O.N.; Tovbis, E.M.; Kazakovtsev, L.A. Optimization of Subgradient Method Parameters Based on Rank-Two Correction of Metric Matrices. *J. Appl. Ind. Math.* **2022**, *16*, 427–439. [CrossRef]
28. Krutikov, V.; Gutova, S.; Tovbis, E.; Kazakovtsev, L.; Semenkin, E. Relaxation Subgradient Algorithms with Machine Learning Procedures. *Mathematics* **2022**, *10*, 3959. [CrossRef]
29. Krutikov, V.; Tovbis, E.; Stanimirović, P.; Kazakovtsev, L. On the Convergence Rate of Quasi-Newton Methods on Strongly Convex Functions with Lipschitz Gradient. *Mathematics* **2023**, *11*, 4715. [CrossRef]
30. Shor, N.Z. Application of the gradient descent method for solving network transportation problems. In *Scientific Seminar on Theoretic and Applied Problems of Cybernetics and Operations Research*; Nauch. Sovet po Kibernetike Akad. Nauk: Kiev, Ukraine, 1962; pp. 9–17.
31. Polyak, B. A general method for solving extremum problems. *Sov. Math. Dokl.* **1967**, *8*, 593–597.
32. Gol'shtein, E.G.; Nemirovskii, A.S.; Nesterov, Y.E. The level method and its generalizations and applications. *Ekon. Mat. Metody* **1983**, *31*, 164–180. (In Russian)
33. Nesterov, Y. Universal gradient methods for convex optimization problems. *Math. Program. Ser. A.* **2015**, *152*, 381–404. [CrossRef]
34. Gasnikov, A.V.; Nesterov, Y.E. Universal Method for Stochastic Composite Optimization. *arXiv* **2016**, arXiv:1604.05275. [CrossRef]
35. Nemirovskii, A.S.; Yudin, D.B. *Complexity of Problems and Efficiency of Methods in Optimization*; Nauka: Moscow, Russia, 1979.
36. Shor, N. *Minimization Methods for Nondifferentiable Functions*; Springer: Berlin, Germany, 1985.
37. Polyak, B.T. Minimization of nonsmooth functional. *Zh. Vychisl. Mat. Mat. Fiz.* **1969**, *9*, 509–521.
38. Krutikov, V.N.; Samoilenko, N.S.; Meshechkin, V.V. On the Properties of the Method of Minimization for Convex Functions with Relaxation on the Distance to Extremum. *Autom. Remote Contro* **2019**, *80*, 102–111. [CrossRef]
39. Wolfe, P. Note on a method of conjugate subgradients for minimizing nondifferentiable functions. *Math. Program.* **1974**, *7*, 380–383. [CrossRef]
40. Lemarechal, C. An extension of Davidon methods to non-differentiable problems. *Math. Program. Study* **1975**, *3*, 95–109.
41. Dem'yanov, V.F.; Vasil'ev, L.V. *Non-Differentiable Optimization*; Nauka: Moscow, Russia, 1981. (In Russian)
42. Skokov, V.A. Note on minimization methods employing space stretching. *Cybern. Syst. Anal.* **1974**, *10*, 689–692. [CrossRef]
43. Krutikov, V.N.; Gorskaya, T.A. A family of subgradient relaxation methods with rank 2 correction of metric matrices. *Ekon. Mat. Metody* **2009**, *45*, 37–80.

44. Tsypkin, Y.Z. *Foundations of the Theory of Learning Systems*; Academic Press: New York, NY, USA, 1973.
45. Nurminsky, E.A.; Tien, D. Method of conjugate subgradients with constrained memory. *Autom. Remote Control* **2014**, *75*, 646–656. [CrossRef]
46. Bunday, B.D. *Basic Optimization Methods*; Edward Arnold: London, UK, 1984.