M. S. Alkousa, A. V. Gasnikov, E. L. Gladin, I. A. Kuruzov, D. A. Pasechnyuk, F. S. Stonyakin, Solving strongly convex-concave composite saddle-point problems with low dimension of one group of variable, *Sbornik: Mathematics*, 2023, Volume 214, Issue 3, 285–333

DOI: 10.4213/sm9700e

DOI: https://doi.org/10.4213/sm9700e

Solving strongly convex-concave composite saddle-point problems with low dimension of one group of variables

M. S. Alkousa, A. V. Gasnikov, E. L. Gladin,
I. A. Kuruzov, D. A. Pasechnyuk and F. S. Stonyakin

Abstract. Algorithmic methods are developed that guarantee efficient complexity estimates for strongly convex-concave saddle-point problems in the case when one group of variables has a high dimension, while another has a rather low dimension (up to 100). These methods are based on reducing problems of this type to the minimization (maximization) problem for a convex (concave) functional with respect to one of the variables such that an approximate value of the gradient at an arbitrary point can be obtained with the required accuracy using an auxiliary optimization subproblem with respect to the other variable. It is proposed to use the ellipsoid method and Vaidya's method for low-dimensional problems and accelerated gradient methods with inexact information about the gradient or subgradient for high-dimensional problems. In the case when one group of variables, ranging over a hypercube, has a very low dimension (up to five), another proposed approach to strongly convex-concave saddle-point problems is rather efficient. This approach is based on a new version of a multidimensional analogue of Nesterov's method on a square (the multidimensional dichotomy method) with the possibility to use inexact values of the gradient of the objective functional.

Bibliography: 28 titles.

Keywords: saddle-point problem, ellipsoid method, Vaidya's method, inexact subgradient, hypercube, multidimensional dichotomy.

§ 1. Introduction

Saddle-point problems are quite topical since they arise in real-life problems of machine learning, computer graphics, game theory and optimal transport theory. In view of the importance of this type of problem, there are many papers devoted to various algorithms for their solution and theoretical results concerning their rates of convergence (or complexity); see [1]–[8].

The research by A. V. Gasnikov in §§ 2.2 and 3 was carried out within the state assignment of the Ministry of Science and Higher Education of the Russian Federation (project no. 0714-2020-0005). The research by F. S. Stonyakin in §§ 2.1 and 2.3 was supported by the Federal Academic Leadership Programme "Priority 2030" (agreement no. 075-02-2021-1316 of September 30, 2021).

AMS 2020 *Mathematics Subject Classification*. Primary 90C47; Secondary 68Q25, 90C06.

This paper considers convex-concave saddle-point problems of the form

$$\min_{x\in Q_x}\max_{y\in Q_y}\bigl\{\widehat{S}(x,y):=r(x)+F(x,y)-h(y)\bigr\}, \tag{1.1}$$

where $Q_x\subseteq\mathbb{R}^n$ and $Q_y\subseteq\mathbb{R}^m$ are nonempty convex compact sets and $r\colon Q_x\to\mathbb{R}$ and $h\colon Q_y\to\mathbb{R}$ are a μ_x-strongly convex function and a μ_y-strongly convex function, respectively. The functional $F\colon Q_x\times Q_y\to\mathbb{R}$ is convex with respect to x and concave with respect to y and is defined in a neighbourhood of the set $Q_x\times Q_y$. When the problem is not strongly convex (the case $\mu_x=0$ or $\mu_y=0$), we can reduce it to a strongly convex one using regularization methods (see [9], Remark 4.1).

The class of problems (1.1) was studied already in sufficient detail some time ago in the bilinear case, that is, when $F(x,y)=\langle Ax,y\rangle$ for some linear operator A (see, for example, an overview in [6]). Other investigations were aimed at generalizing the results known in the bilinear case to the general situation; see [3], [5], [10] and [11].

In [12] the problem was considered in the case when $Q_x\equiv\mathbb{R}^n$, $Q_y\equiv\mathbb{R}^m$, and for any x and y the function $\widehat{S}(x,y)=F(x,y)$ is μ_x-strongly convex with respect to x, μ_y-strongly concave with respect y and (L_{xx},L_{xy},L_{yy})-smooth. The last property means that for any fixed x the maps $\nabla_yF(x,\,\cdot\,)$ and $\nabla_xF(x,\,\cdot\,)$ are Lipschitz-continuous with some nonnegative constants L_{yy} and L_{xy}, while for any fixed y the maps $\nabla_xF(\cdot,y)$ and $\nabla_yF(\cdot,y)$ are Lipschitz-continuous with constants L_{xx} and L_{xy}. In [12], for this class of problems a lower complexity estimate of the form

$$N(\varepsilon)=\Omega\biggl(\sqrt{\frac{L_{xx}}{\mu_x}+\frac{L_{xy}^2}{\mu_x\mu_y}+\frac{L_{yy}}{\mu_y}}\,\ln\biggl(\frac{1}{\varepsilon}\biggr)\biggr)$$

was substantiated, where $N(\varepsilon)=\Omega(f(\varepsilon))$ means that there are $C>0$ and $\varepsilon_0>0$ such that $|N(\varepsilon)|>C|f(\varepsilon)|$ for any $\varepsilon<\varepsilon_0$. In [1], an approach based on accelerated methods was proposed with a complexity estimate which was closest to the optimal one at that time. Thereafter, attempts were made to obtain an optimal algorithm; see [4] and [7]. For instance, a method proposed in [13] had an upper estimate of the form $\widetilde{O}\bigl(\sqrt{L_{xx}/\mu_x+L\cdot L_{xy}/(\mu_x\mu_y)+L_{yy}/\mu_y}\,\bigr)$, where $L=\max\bigl\{L_{xx},L_{xy},L_{yy}\bigr\}$ for the number of iterations (the notation $\widetilde{O}(\,\cdot\,)$ means $O(\,\cdot\,)$ up to a factor logarithmic in ε^{-1} and raised to power 1 or 2). Thus, the problem of a near-optimal algorithm for a strongly convex-concave saddle-point problem of high dimension with smooth objective function was solved.

Table 1 shows the best currently known results (see [1]–[7] and the references there) concerning complexity estimates for the solution of problem (1.1).

In each case, an ε-solution of (1.1) can be obtained after $\widetilde{O}(\,\cdot\,)$ (specified in the first column) iterations of the quantity in the second column. The Lipschitz constant for ∇F (the gradient with respect to x and y) is denoted by L_F. Now, a function $r\colon Q_x\to\mathbb{R}$ is said to be prox-friendly if we can solve explicitly a problem of the form

$$\min_{x\in Q_x}\bigl\{\langle c_1,x\rangle+r(x)+c_2\|x\|_2^2\bigr\},\qquad c_1\in Q_x,\quad c_2>0. \tag{1.2}$$

The prox-friendliness of $h\colon Q_y\to\mathbb{R}$ is defined similarly for problems of the form

$$\min_{y\in Q_y}\bigl\{\langle c_3,y\rangle+h(y)+c_4\|y\|_2^2\bigr\},\qquad c_3\in Q_y,\quad c_4>0. \tag{1.3}$$

Table 1. Best known results on the complexity of methods for the problem (1.1)

Estimate	Calls of subroutines
Case 1: both functions r and h are prox-friendly	
$\widetilde{O}\Big(\frac{L_F}{\sqrt{\mu_x\mu_y}}\Big)$	computing $\nabla_x F(x,y)$ and $\nabla_y F(x,y)$ and solving problems (1.2) and (1.3)
Case 2: r is an L_x-smooth not prox-friendly function	
$\widetilde{O}\Big(\sqrt{\frac{L_x L_F}{\mu_x\mu_y}}\Big)$	computing $\nabla r(x)$
$\widetilde{O}\Big(\frac{L_F}{\sqrt{\mu_x\mu_y}}\Big)$	computing $\nabla_x F(x,y)$ and $\nabla_y F(x,y)$ and solving problem (1.3)
Case 3: h is an L_y-smooth not prox-friendly function	
$\widetilde{O}\Big(\sqrt{\frac{L_y L_F}{\mu_x\mu_y}}\Big)$	computing $\nabla h(y)$
$\widetilde{O}\Big(\frac{L_F}{\sqrt{\mu_x\mu_y}}\Big)$	computing $\nabla_x F(x,y)$ and $\nabla_y F(x,y)$ and solving problem (1.2)
Case 4: r and h are L_x- and L_y-smooth not prox-friendly functions	
$\widetilde{O}\Big(\sqrt{\frac{L_x L_F}{\mu_x\mu_y}}\Big)$	computing $\nabla r(x)$
$\widetilde{O}\Big(\sqrt{\frac{L_y L_F}{\mu_x\mu_y}}\Big)$	computing $\nabla h(y)$
$\widetilde{O}\Big(\frac{L_F}{\sqrt{\mu_x\mu_y}}\Big)$	computing $\nabla_x F(x,y)$ and $\nabla_y F(x,y)$

Note that a saddle-point problem can also be reduced to a variational inequality with monotone operator. Recall that an operator $G\colon \operatorname{dom} G \to \mathbb{R}^k$ defined on a convex set $\operatorname{dom} G \subseteq \mathbb{R}^k$ is called *monotone* if

$$\langle G(z) - G(z'),\ z - z'\rangle \geqslant 0 \quad \forall\, z, z' \in Q,$$

where Q is a convex compact set with nonempty interior and $\operatorname{int} Q \subseteq \operatorname{dom} G$. A *solution of a variational inequality* is a point $z_* \in \operatorname{dom} G \cap Q$ such that

$$\langle G(z_*),\ z - z_*\rangle \geqslant 0 \quad \forall\, z \in Q.$$

Convex-concave saddle-point problems of the form (1.2) with differentiable function $S(\,\cdot\,,\cdot\,)$ and functions r and h identically equal to zero are reducible to a variational inequality with operator $G(x,y) = [\nabla_x S(x,y), -\nabla_y S(x,y)]^\top$, which is monotone because S is convex in x and concave in y. A variational inequality of this type can be solved, for example, using the ellipsoid method from [14], which yields the rate of convergence $O\left(\exp\left\{-\dfrac{N}{2d(d+1)}\right\}\right)$, where $d=n+m$ is the dimension of the problem and N is the number of iterations. This approach to the solution of problem (1.2) requires neither the smoothness nor the strong convexity (concavity) of the objective function $S(\,\cdot\,,\cdot\,)$ and can be considered to be quite efficient in the case when the dimension of the problem $d=n+m$ is low. In addition, based on methods like the centre of gravity method, we can improve the estimate $O\left(\exp\left\{-\dfrac{N}{2d(d+1)}\right\}\right)$ to $O\left(\exp\left\{-\dfrac{N}{O(d)}\right\}\right)$; see [15], Lecture 5. Note that the first of the above approaches to problem (1.1) is quite efficient in the case when both variables in (1.1) are of high dimension, whereas the second approach is efficient when the dimensions of the variables in the problem are low.

This paper considers the situation when one group of variables (x or y) has a high dimension, while the other has a low dimension (of several dozens). Problem (1.1) is represented as a minimization problem with respect to the 'outer' variable of a convex function such that information about it (values of the function or its gradient) is only available with some accuracy. This accuracy, in its turn, is controlled using an auxiliary optimization subproblem with respect to the 'inner' variable. Accordingly, depending on the (low or high) dimension of the outer variable x, it is natural to distinguish two approaches of this kind. If the dimension of the outer variable x is high (§ 2.2), then it is proposed to use the accelerated gradient method with inexact oracle to solve (1.1) and a cutting plane method (the ellipsoid or Vaidya's method) to solve the auxiliary maximization subproblem. If the dimension of the outer variable x is low, then the versions of cutting plane methods (the ellipsoid and Vaidya's method) proposed in this paper are applied to solve (1.1) using inexact analogues of the gradient of the objective function (δ-subgradients or δ-additively inexact gradients) in iterations, while accelerated gradient methods are applied to solve the inner optimization subproblem. To deduce estimates for the number of iterations (calls of the subroutine for finding the (sub)gradient of a functional or its 'inexact analogue') sufficient for attaining the required quality of solutions of the saddle-point problem (1.1) using the above approach, important theoretical results are obtained that describe the effect of the parameter δ on the quality of the exit point of the ellipsoid or Vaidya's method using δ-subgradients or δ-additively inexact gradients instead of the gradient of the objective function in iterations. Note that Vaidya's method, in comparison with the ellipsoid method, yields a better estimate for the number of iterations sufficient the necessary quality of the approximate solution; however, the cost per iteration in the ellipsoid method is lower. Thereafter, we consider in detail the situation when the dimension of one group of variables in the saddle-point problem (1.1) is very low and the feasible set of values of this variable is a hypercube. In this case it is possible to use an analogue of the dichotomy method instead of the ellipsoid method, which can turn out to be more profitable in the case of a very low dimension (up to five) than cutting plane methods. More precisely, this paper proposes a minimization method for a convex differentiable function with Lipschitz gradient on a multidimensional hypercube for low dimension of the space, which is an analogue of Nesterov's minimization method for a convex Lipschitz-continuous function of two variables on a square with fixed side (see [9] and [16]). In what follows we call this method the *multidimensional dichotomy method*. The idea of this method is to divide the square into smaller parts and remove them gradually so that all values of the objective function in the remaining rather small domain are sufficiently close to the optimal value. The method consists in solving auxiliary one-dimensional minimization problems along separating line segments and does not involve computing the exact value of the gradient of the objective functional (that is, the method can be regarded as an incomplete gradient method). In the two-dimensional case on a square this method was considered in [16]. Our paper proposes a new version of the stopping criterion for the auxiliary subproblem and also an analogue of Nesterov's method for an arbitrary dimension. The results obtained also apply to saddle-point problems with low dimension of one group of variables on a hypercube. Complexity estimates are given for this approach to strongly convex-concave saddle-point problems of the form (1.1) with sufficiently smooth functionals in the case when the low-dimensional problem is solved by the dichotomy method with inexact gradient and a high-dimensional auxiliary problem is solved at each iteration using the fast gradient method for all auxiliary problems. The estimates obtained for the rate of convergence seem to be acceptable if the dimension of one group of variables is sufficiently low (up to five).

To compare the proposed approaches to the problem (1.1), one with another and also with some known analogues, we performed computational experiments for some types of Lagrangian saddle-point problems associated to minimization problems for strongly convex functionals with quite a few convex functional inequality constraints. In particular, numerical experiments were carried out for the dual problem of the LogSumExp problem with linear constraints (applications of problems of this type are described, for example, in [2]). We have compared the running speeds of the approach to (1.1) as a system of subproblems in the cases when the primal low-dimensional problem is solved by the fast gradient method with (δ, L)-oracle or by the ellipsoid method with δ-subgradient. The computational experiments mentioned above showed that the use of the cutting plane methods under consideration in low-dimensional problems results in a more successful work with a rather high required accuracy in comparison with methods involving only gradient approaches. In addition, the multidimensional dichotomy method, which we propose here, has proved to be more efficient in some cases than cutting plane methods, which indicates that using this approach can also be reasonable. In the case when the low-dimensional subproblem is solved by the ellipsoid method, we made additional experiments to compare different ways of taking account of inexactness when gradients with additive noise are used (§ 3.4): using (1.1) as a system of subproblems the proposed approach, with inexactness values varying with the diameter of the current ellipsoid, makes it possible to attain the stopping condition and thus the prescribed accuracy more rapidly. We also made an experiment to compare the efficiency of the proposed approaches with a known analogue [8] on the problem of projecting a point onto a set specified by a system of (a small number of) smooth constraints (§ 3.5). The comparison has shown that the approach using the ellipsoid method for the low-dimensional problem with the proposed stopping condition and new estimates for the sufficient number of iterations for the inner method turns out to be much more efficient than the analogous approach in [8]. Note that in cutting plane methods or the multidimensional analogue of the dichotomy method that are applied in our paper to low-dimensional subproblems strong convexity is only important for theoretical estimates. To implement these methods it is not necessary to assume the strong convexity of the objective function; therefore, we do not regularize the Lagrangian saddle-point problems with respect to the dual variables.

This paper contains an introduction, two basic sections and conclusions. In § 2 the main results and approaches to the problem under consideration are presented for various cases of low dimension of the outer and inner variables. Subsection 2.1 describes the general scheme of reasoning used to analyze the problems treated in this paper, which involves considering a family of auxiliary optimization subproblems. Subsection 2.2 is a key one; it contains the derivation of estimates for problem (1.1) in the case of a relatively low dimension of one group of variables on the basis of new variants of the ellipsoid method and Vaidya's method for the corresponding auxiliary subproblems. Subsection 2.3 considers the special case of a very low (up to five) dimension of one group of variables in the saddle-point problem and produces complexity estimates for the approach to the problem (1.1) based on the proposed multidimensional analogue of the dichotomy method with additively inexact gradient. In § 3 the results of some computational experiments are presented and the running speeds of the proposed approaches are compared. The complete proofs of some results (Theorems 3, 6–9 and Lemma 4) are given in § 4.

§ 2. Basic results

2.1. Scheme for the derivation of complexity estimates for the class of saddle-point problems under consideration.

2.1.1. *The statement of the problem.* We rewrite problem (1.1) as

$$\min_{x\in Q_x}\Bigl\{g(x):=r(x)+\max_{y\in Q_y}S(x,y)\Bigr\}, \tag{2.1}$$

where $Q_x\subseteq\mathbb{R}^n$ and $Q_y\subseteq\mathbb{R}^m$ are convex closed sets, Q_x is bounded, and $S(x,y)=F(x,y)-h(y)$ is a continuous function in (1.1) that is strongly convex with respect to x and strongly concave with respect to y.

Definition 1. A pair of points $(\widetilde{x},\widetilde{y})\in Q_x\times Q_y$ is called an *ε-solution* of the problem (1.1) (or (2.1)) if

$$\max\{\|\widetilde{x}-x_*\|_2,\ \|\widetilde{y}-y_*\|_2\}\leqslant\varepsilon, \tag{2.2}$$

where (x_*,y_*) is an exact solution of the strongly convex-concave saddle-point problem (1.1).

Remark 1. In view of the strong convexity of g, for (2.2) to hold it suffices to find $\widetilde{x}$ in problem (2.1) such that $g(\widetilde{x})-\min_{x\in Q_x}g(x)\leqslant C\varepsilon^2$ (for an appropriate choice of the positive constant C depending on the strong convexity parameter μ_x) and also to solve the auxiliary subproblems with a similar accuracy of $O(\varepsilon^2)$. Since we use methods guaranteeing a linear rate of convergence, the final complexity estimates for (1.1) contain quantities like $C\varepsilon^2$ under the sign of logarithm. That is, to deduce asymptotic complexity estimates for saddle-point problems of the form (1.1), it suffices to find $\widetilde{x}$ such that $g(\widetilde{x})-\min_{x\in Q_x}g(x)\leqslant\varepsilon$. We use this in our reasoning.

We view (2.1) as the composition of the inner maximization problem

$$\widehat{g}(x) := \max_{y \in Q_y} S(x, y) \tag{2.3}$$

and the outer minimization problem

$$\min_{x \in Q_x} g(x). \tag{2.4}$$

The iteration method for the outer problem (2.4) uses the gradient of the objective function at each step, which can be computed with some accuracy based on an approximate solution of the inner problem (2.3). In this connection we need clear estimates for the quality of solutions produced by our method in the case when inexact information about the gradient or subgradient of the objective function is used in iterations. It turns out that δ-subgradients (see Definition 2), (δ, L)-subgradients (see (2.5) below and also [1]) and δ-inexact subgradients (see Definition 3 below) can be regarded as an appropriate inexact analogue of the subgradient of the objective function for saddle-point problems.

Definition 2. Let $\delta \geqslant 0$. A vector $\nu(\widehat{x}) \in \mathbb{R}^n$ is said to be a δ-*subgradient* of a convex function $g\colon Q_x \to \mathbb{R}$ at a point $\widehat{x}$ if $g(x) \geqslant g(\widehat{x}) + \langle \nu(\widehat{x}), x - \widehat{x} \rangle - \delta$ for any $x \in Q_x$. The set of δ-subgradients of g at $\widehat{x}$ is denoted by $\partial_\delta g(\widehat{x})$.

Note that δ-subgradients coincide with the ordinary subgradient for $\delta = 0$.

Definition 3. Let $\delta \geqslant 0$. A vector $\nu(\widehat{x}) \in \mathbb{R}^n$ is said to be a δ-*inexact subgradient* of a convex function $g\colon Q_x \to \mathbb{R}$ at a point $\widehat{x}$ if $\|\nabla g(\widehat{x}) - \nu(\widehat{x})\|_2 \leqslant \delta$ for some subgradient $\nabla g(\widehat{x}) \in \partial g(\widehat{x})$. If we know that g is differentiable at $\widehat{x}$, then $\nu(\widehat{x})$ is said to be a δ-*inexact gradient.*

2.1.2. *Computing inexact analogues of the gradient of the objective function in the primal subproblem.* As an approximate subgradient of the objective function g in the problem (2.4) at a point $x \in Q_x$, we propose to use the subgradient $\nabla r(x) + \nabla_x S(x, \widetilde{y})$, where $\widetilde{y}$ is an $\widetilde{\varepsilon}$-solution of the auxiliary subproblem (2.3) for x, and $\nabla r(x) \in \partial r(x)$ and $\nabla_x S(x, \widetilde{y}) \in \partial_x S(x, \widetilde{y})$ are arbitrary finite subgradients at x of r and $S(\cdot, \widetilde{y})$, respectively. It turns out that an inexact subgradient of this kind can be a δ-subgradient of the objective function g at x if the accuracy $\widetilde{\varepsilon}$ for the auxiliary problem is chosen according to the following lemma.

Lemma 1 (see [17], pp. 123, 124). *For fixed x assume that $\widetilde{y} \in Q_y$ in problem* (2.1) *is such that $\widehat{g}(x) - S(x, \widetilde{y}) \leqslant \delta$. Then $\partial_x S(x, \widetilde{y}) \subseteq \partial_\delta(\widehat{g}(x))$.*

This lemma states that to find a δ-subgradient of the function g it suffices to solve the maximization problem (2.3) with accuracy $\widetilde{\varepsilon} = \delta$.

It turns out (see [1]) that for saddle-point problems of the form (2.1) with the corresponding assumptions and accuracy of solving the auxiliary problem, we can guarantee that a (δ, L)-subgradient $\nabla_{\delta,L} g(x)$ of g at an arbitrary point $x \in Q_x$ can be found for an appropriate $L > 0$ and an arbitrarily small $\delta > 0$:

$$g(x) + \langle \nabla_{\delta,L} g(x), y - x \rangle - \delta \leqslant g(y) \leqslant g(x) + \langle \nabla_{\delta,L} g(x), y - x \rangle + \frac{L}{2}\|y - x\|_2^2 + \delta. \tag{2.5}$$

It is clear that a (δ, L)-subgradient $\nabla_{\delta,L} g(x)$ is a δ-subgradient of g at x with an additional condition of the form (2.5).

The following known result is implied directly by a similar result [18] for the well-known notion of a (δ, L)-oracle and explains the relationship, which we use in what follows, between two analogues of the gradient (δ-subgradients and δ-inexact subgradients) of a convex function g admitting a (δ, L)-Lipschitz subgradient at each point $x \in Q_x$.

Theorem 1. *Let $g\colon Q_x \to \mathbb{R}$ be a convex function, and let $\nu(x) = \nabla_{\delta,L} g(x)$ be a (δ, L)-subgradient of it at a point $x \in \operatorname{int} Q_x$. Let $\rho(x, \partial Q_x)$ be the Euclidean distance of x to the boundary of the set Q_x. If $\rho(x, \partial Q_x) \geqslant 2\sqrt{\delta/L}$, then for any subgradient $\nabla g(x)$*

$$\|\nu(x) - \nabla g(x)\|_2 \leqslant 2\sqrt{\delta L}.$$

From Lemma 1 and Theorem 1 we can conclude that to find a δ-inexact subgradient of g at a point $x \in \operatorname{int} Q_x$ for sufficiently small $\delta > 0$ it suffices to solve the maximization problem (2.3) with accuracy $\widetilde{\varepsilon} = \delta^2/(4L)$.

Another method for finding a δ-inexact subgradient (in this case, a δ-inexact gradient, since differentiability is additionally assumed) can be used under the following additional assumption.

Assumption 1. The function $S(\cdot, y)$ is differentiable for all $y \in Q_y$ and satisfies the condition

$$\|\nabla_x S(x, y) - \nabla_x S(x, y')\|_2 \leqslant L_{xy} \|y - y'\|_2 \quad \forall\, x \in Q_x, \quad y, y' \in Q_y, \tag{2.6}$$

for some $L_{xy} \geqslant 0$.

Lemma 2. *Under the assumptions of problem (2.1) and Assumption 1, for any fixed $x \in Q_x$, let $\widetilde{y} \in Q_y$ be a point such that $\widehat{g}(x) - S(x, \widetilde{y}) \leqslant \widetilde{\varepsilon}$. Then $\widehat{g}$ is differentiable at x and*

$$\|\nabla_x S(x, \widetilde{y}) - \nabla \widehat{g}(x)\|_2 \leqslant L_{xy} \sqrt{\frac{2\widetilde{\varepsilon}}{\mu_y}}.$$

Thus, to find a δ-inexact gradient of the function g it suffices to solve the maximization problem (2.3) with accuracy

$$\widetilde{\varepsilon} = \frac{\mu_y}{2L_{xy}^2} \delta^2$$

if $L_{xy} > 0$ and with an arbitrary finite accuracy $\widetilde{\varepsilon}$ if $L_{xy} = 0$.

We state another lemma concerning the relationship between two analogues of the gradient (δ-subgradients and δ-inexact subgradients), which are mentioned below in this paper.

Lemma 3. *Let $g\colon Q \to \mathbb{R}$ be a convex function on a convex set Q. Then the following hold:*

1) *if Q is bounded, then a δ_1-inexact subgradient of g is a δ_2-subgradient of g, where*

$$\delta_2 = \delta_1 \operatorname{diam} Q \quad \text{and} \quad \operatorname{diam} Q = \sup_{x, x' \in Q} \|x - x'\|_2;$$

2) *if the function g is μ-strongly convex, then a δ_1-inexact subgradient of g is a δ_2-subgradient with $\delta_2 = \delta_1^2/(2\mu)$.*

The methods of research and the results in this paper can conventionally be divided into two groups such that the first group (§ 2.2) is related to cutting plane methods (the ellipsoid method and Vaidya's method) for subproblems of low dimension and the second (§ 2.3) uses our multidimensional version of the dichotomy method with adaptive stopping rules for subproblems of low dimension (approximately, up to five). For results based on cutting plane methods for primal subproblems the above assertions make it possible to substantiate the potential use of both δ-subgradients and δ-inexact subgradients of the objective function g in iterations. The complexity estimates for saddle-point problems coincide asymptotically in these cases. As for the second group of results, which is related to the multidimensional dichotomy method, the assumption of the smoothness of the function and its δ-inexact gradient (which is precisely the gradient, since this part of the paper considers only smooth problems) is essential there.

2.1.3. *The general scheme* (*algorithm*) *of the approach to the class of problems selected.* In this subsection, we present an algorithm for the min-max problem (2.1), while the following subsections consider specific examples of methods used in the general algorithm and the corresponding complexity estimates.

Algorithm 1 (algorithm for the min-max problem (2.1)).

```
Require: a method $\mathcal{M}_1$ for the solution of problem (2.4) using $\delta$-subgradients or $\delta$-inexact gradients, the number $N > 0$ of steps in this method, a method $\mathcal{M}_2$ for the solution of problem (2.3), the accuracy $\widetilde{\varepsilon}$ of its solution, the initial approximation $(x^0, y^0)$.
for $k = 0, \dots, N-1$ do
    Solve problem (2.3) for fixed $x = x^k$ with accuracy $\widetilde{\varepsilon}$ using the method $\mathcal{M}_2$ starting from $y^k$:
        $y^{k+1} := \mathcal{M}_2(x^k, y^k, \widetilde{\varepsilon})$.
    Set $\nu^{k+1} := \nabla r(x^k) + \nabla_x S(x^k, y^{k+1}) \in \partial r(x^k) + \partial_x S(x^k, y^{k+1})$.
    Make one step of the method $\mathcal{M}_1$ from the point $x^k$ using the approximate gradient $\nu^{k+1}$:
        $x^{k+1} := \mathrm{step}(\mathcal{M}_1, x^k, \nu^{k+1})$.
end for
Ensure: $x^N$.
```

We find the complexity of Algorithm 1 in accordance with the following obvious principle.

Proposition. *Assume that a method $\mathcal{M}_1$ for the solution of problem* (2.4) *using δ-subgradients or δ-inexact gradients finds an ε-solution after at most $N_1(\varepsilon, \delta)$ steps,*[1] *and assume that a method $\mathcal{M}_2$ for the solution of problem* (2.3) *finds an $\widetilde{\varepsilon}$-solution after at most $N_2(\widetilde{\varepsilon})$ steps. If the accuracy δ of the oracle for problem* (2.4) *depends on $\widetilde{\varepsilon}$ as $\delta(\widetilde{\varepsilon})$, then Algorithm 1 finds an ε-solution of problem* (2.1) *after $N_1(\varepsilon, \delta(\widetilde{\varepsilon}))$ iterations of $\nabla_x S$ and $N_1(\varepsilon, \delta(\widetilde{\varepsilon})) \cdot N_2(\widetilde{\varepsilon})$ iterations of $\nabla_y S$.*

[1] For a particular method $\mathcal{M}_1$ to guarantee accuracy ε after a finite number of step, we can need δ to be sufficiently small in comparison with ε (for example, $\delta < \varepsilon$). It is assumed in the proposition that this condition holds.

2.2. Cutting plane methods using inexact analogues of the subgradient and their applications to complexity estimates for saddle-point problems with low dimension of one group of variables. We describe specific methods that can be used in Algorithm 1 in the case when the dimension of the outer or inner variable is relatively low (at most 100) and the objective function is of composite structure.

We start by stating the problem and describing the methods briefly; we also deduce complexity estimates in the case of a low dimension of the primal subproblem (in other words, of the outer variable).

Assume that the following holds for problem (2.1).

Assumption 2. The set Q_x has a nonempty interior, the dimension n is relatively low (at most 100), $Q_y \equiv \mathbb{R}^m$, and the function S has the form

$$S(x,y) := F(x,y) - h(y),$$

where the μ_y-strongly convex function h is continuous and the convex-concave function F is differentiable with respect to y and satisfies

$$\|\nabla_y F(x,y) - \nabla_y F(x,y')\|_2 \leqslant L_{yy}\|y-y'\|_2 \quad \forall\, x \in Q_x, \quad y,y' \in Q_y,$$

for some $L_{yy} \geqslant 0$.

Assume that one of the following conditions is also satisfied:

a) h is prox-friendly, that is, we can explicitly solve the problem

$$\min_{y\in Q_y}\bigl\{\langle c_1,y\rangle + h(y) + c_2\|y\|_2^2\bigr\}, \qquad c_1 \in Q_y, \quad c_2 > 0; \tag{2.7}$$

b) h has an L_h-Lipschitz gradient.

Theorem 2. *An ε-solution of problem* (2.1) *can be attained*:

- *under Assumption* 2, *after*

$$O\biggl(n\ln\frac{n}{\varepsilon}\biggr) \quad \textit{rounds of calculation of} \ \ \nabla_x F, \ \nabla r;$$

- *under Assumption* 2,a), *after*

$$O\biggl(n\sqrt{\frac{L_{yy}}{\mu_y}}\ln\frac{n}{\varepsilon}\ln\frac{1}{\varepsilon}\biggr) \quad \textit{rounds of calculation of} \ \ \nabla_y F \ \textit{and solutions of problem (2.7)};$$

- *under Assumption* 2,b), *after*

$$O\biggl(n\sqrt{\frac{L_{yy}}{\mu_y}}\ln\frac{n}{\varepsilon}\ln\frac{1}{\varepsilon}\biggr) \quad \textit{rounds of calculation of} \ \ \nabla_y F \ \textit{and}$$

$$O\biggl(n\sqrt{\frac{L_h}{\mu_y}}\ln\frac{n}{\varepsilon}\ln\frac{1}{\varepsilon}\biggr) \quad \textit{rounds of calculation of} \ \ \nabla h.$$

Below we describe methods that apply to the auxiliary subproblems in Algorithm 1 and also theoretical results concerning convergence rate estimates.

2.2.1. *Cutting plane methods using δ-subgradients.* We consider a problem of the form

$$\min_{x \in Q} g(x), \tag{2.8}$$

where $Q \subset \mathbb{R}^n$ is a convex compact set that lies in a Euclidean ball of radius $\mathcal{R}$ and contains a Euclidean ball of radius $\rho > 0$, g is a continuous convex function, and a positive number B is such that

$$|g(x) - g(x')| \leqslant B \quad \forall\, x, x' \in Q.$$

We propose a generalization of the ellipsoid method (Algorithm 2) for problem (2.8) that uses δ-subgradients of the objective function in iterations.

Algorithm 2 (ellipsoid method with δ-subgradient for the problem (2.8)).

```
Require: the number of iterations $N > 0$, $\delta \geqslant 0$, a ball $\mathcal{B}_{\mathcal{R}} \supseteq Q$, its centre $c$
    and radius $\mathcal{R}$.
$\mathcal{E}_0 := \mathcal{B}_{\mathcal{R}}$, $H_0 := \mathcal{R}^2 I_n$, $c_0 := c$.
for $k = 0, \dots, N-1$ do
  if $c_k \in Q$ then
    $w_k := w \in \partial_\delta g(c_k)$,
    if $w_k = 0$ then
      return $c_k$,
    end if
  else
    $w_k := w$, where $w \neq 0$ is such that $Q \subset \{x \in \mathcal{E}_k \colon \langle w, x - c_k \rangle \leqslant 0\}$.
  end if
  $c_{k+1} := c_k - \frac{1}{n+1} \frac{H_k w_k}{\sqrt{w_k^\top H_k w_k}}$,
  $H_{k+1} := \frac{n^2}{n^2 - 1} \left( H_k - \frac{2}{n+1} \frac{H_k w_k w_k^\top H_k}{w_k^\top H_k w_k} \right)$,
  $\mathcal{E}_{k+1} := \{x \colon (x - c_{k+1})^\top H_{k+1}^{-1} (x - c_{k+1}) \leqslant 1\}$,
end for
Ensure: $x^N = \arg\min_{x \in \{c_0, \dots, c_N\} \cap Q} g(x)$.
```

Theorem 3 (quality estimate for an approximate solution for the ellipsoid method using δ-subgradients). *The point $x^N \in Q$ obtained after $N \geqslant 2n^2 \ln(\mathcal{R}/\rho)$ iterations of Algorithm 2 for problem (2.8) satisfies the inequality*

$$g(x^N) - \min_{x \in Q} g(x) \leqslant \frac{B\mathcal{R}}{\rho} \exp\biggl(-\frac{N}{2n^2}\biggr) + \delta. \tag{2.9}$$

Corollary 1. *If the assumptions of Theorem 3 are supplemented with the condition that g is μ_x-strongly convex, then the output point x^N of Algorithm 2 satisfies the inequality*

$$\|x^N - x_*\|_2^2 \leqslant \frac{2}{\mu_x} \biggl(\frac{B\mathcal{R}}{\rho} \exp\biggl(-\frac{N}{2n^2}\biggr) + \delta \biggr), \tag{2.10}$$

where x_* is the required minimum point of g.

Remark 2. The μ_x-strong convexity of g and estimate (2.10) are essential for the attainability of the required quality of the solution of the saddle-point problem (1.1) in accordance with Definition 1.

Remark 3. The ellipsoid method with δ-subgradient can also be used in the case when a δ-inexact gradient is available instead of the exact gradient or δ-subgradients (see Lemma 3).

Now we recall the cutting plane method proposed by Vaidya (see [19]) to solve the problem (2.8). First we introduce the requisite notation. Given a matrix A and a vector b, we consider an auxiliary bounded n-dimensional polytope $P(A,b)$ of the form

$$P(A,b)=\{x\in\mathbb{R}^n\colon Ax\geqslant b\},\quad \text{where } A\in\mathbb{R}^{m\times n},\ b\in\mathbb{R}^m,$$

where $Ax\geqslant b$ is understood as a componentwise inequality (each coordinate of the vector Ax is not smaller than the corresponding coordinate of b).

We can introduce a logarithmic barrier for the set $P(A,b)$:

$$L(x;A,b)=-\sum_{i=1}^{m}\ln(a_i^\top x-b_i),\qquad x\in\operatorname{int}P(A,b),$$

where $a_i^\top$ is the ith row of the matrix A and $\operatorname{int}P(A,b)$ is the interior of $P(A,b)$. The Hessian H of the function L is

$$H(x;A,b)=\sum_{i=1}^{m}\frac{a_ia_i^\top}{(a_i^\top x-b_i)^2},\qquad x\in\operatorname{int}P(A,b). \tag{2.11}$$

The matrix $H(x;A,b)$ is positive definite for all $x\in\operatorname{int}P(A,b)$. We can also introduce a volumetric barrier for $P(A,b)$:

$$V(x;A,b)=\frac{1}{2}\ln(\det H(x;A,b)),\qquad x\in\operatorname{int}P(A,b), \tag{2.12}$$

where $\det H(x;A,b)$ denotes the determinant of $H(x;A,b)$. We introduce the notation

$$\sigma_i(x;A,b)=\frac{a_i^\top(H(x;A,b))^{-1}a_i}{(a_i^\top x-b_i)^2},\qquad x\in\operatorname{int}P(A,b),\quad 1\leqslant i\leqslant m. \tag{2.13}$$

A *volumetric centre* of the set $P(A,b)$ is a minimum point of the volumetric barrier, that is,

$$x_c=\arg\min_{x\in\operatorname{int}P(A,b)}V(x;A,b). \tag{2.14}$$

The volumetric barrier V is a self-concordant function; thus, it can efficiently be minimized using Newton's method. A detailed theoretical analysis of Vaidya's method can be found in [19] and [20]. It was proved in [21] that Vaidya's method can use δ-subgradients instead of the exact subgradient. Below we present a variant of this method that uses δ-subgradients (Algorithm 3).

Algorithm 3 (Vaidya's method using δ-subgradients for problems of the form (2.8)).

```
Require: the number of iterations $N > 0$, $\delta \geqslant 0$, a pair $(A_0, b_0)$ (see (2.15)),
    $m_0 := n+1$, the parameters $\eta \leqslant 10^{-4}$ and $\gamma \leqslant 10^{-3} \cdot \eta$ of the algorithm.
 1: for $k = 0, \dots, N-1$ do
 2:   Find an approximate volumetric centre, see (2.14).
 3:   Compute $H_k^{-1} := (H(x_k; A_k, b_k))^{-1}$ and $\{\sigma_i(x_k; A_k, b_k)\}_{i=1}^{m_k}$ using
      formulae (2.11) and (2.13),
 4:   $i_k := \arg\min_{1 \leqslant i \leqslant m_k} \sigma_i(x_k; A_k, b_k)$.
 5:   if $\sigma_{i_k}(x_k; A_k, b_k) < \gamma$ then
 6:     Obtain $(A_{k+1}, b_{k+1})$ via eliminating the $i_k$th row in $(A_k, b_k)$,
 7:     $m_{k+1} := m_k - 1$.
 8:   else
 9:     $c_k \in -\partial_\delta g(x_k)$.
10:     Derive $\beta_k \in \mathbb{R}$ such that $c_k^\top x_k \geqslant \beta_k$ from the equation
          $$\frac{c_k^\top H_k^{-1} c_k}{(c_k^\top x_k - \beta_k)^2} = \frac{1}{2}\sqrt{\eta\gamma},$$
11:     $A_{k+1} := \begin{pmatrix} A_k \\ c_k^\top \end{pmatrix}$, $b_{k+1} := \begin{pmatrix} b_k \\ \beta_k \end{pmatrix}$, $m_{k+1} = m_k + 1$.
12:   end if
13: end for
Ensure: $x_N = \arg\min_{x \in \{x_0, \dots, x_{N-1}\}} g(x)$.
```

This algorithm returns a sequence of pairs $(A_k, b_k) \in \mathbb{R}^{m_k \times n} \times \mathbb{R}^{m_k}$ such that the corresponding polyhedra contain the required solution of the problem. As an initial polytope specified by the pair (A_0, b_0) we can choose, for example, the simplex

$$P_0 = \left\{ x \in \mathbb{R}^n \colon x_j \geqslant -\mathcal{R}, j = 1, \dots, n, \sum_{j=1}^{n} x_j \leqslant n\mathcal{R} \right\} \supseteq \mathcal{B}_{\mathcal{R}} \supseteq \mathcal{X},$$

so that

$$b_0 = -\mathcal{R} \begin{bmatrix} \mathbf{1}_n \\ n \end{bmatrix} \quad \text{and} \quad A_0 = \begin{bmatrix} I_n \\ -\mathbf{1}_n^\top \end{bmatrix}, \tag{2.15}$$

where I_n denotes the identity matrix of size $n \times n$ and $\mathbf{1}_n$ denotes the vector $(1, \dots, 1)^\top \in \mathbb{R}^n$. In this case m_0 is equal to $n+1$.

Theorem 4 (see [21]). *After*

$$N \geqslant \frac{2n}{\gamma} \ln\left(\frac{n^{1.5}\mathcal{R}}{\gamma\rho} \right) + \frac{1}{\gamma} \ln \pi$$

iterations Vaidya's method with δ-subgradient for problem (2.8) returns a point x^N such that

$$g(x^N) - \min_{x \in Q} g(x) \leqslant \frac{n^{1.5} B \mathcal{R}}{\gamma\rho} \exp\left(\frac{\ln \pi - \gamma N}{2n} \right) + \delta, \tag{2.16}$$

where $\gamma > 0$ is a parameter of Algorithm 3.

Corollary 2. *If the assumptions in Theorem 4 are supplemented with the μ_x-strong convexity of g, then the output point x^N of Algorithm 3 satisfies the inequality*

$$\|x^N - x_*\|_2^2 \leqslant \frac{2}{\mu_x}\biggl(\frac{n^{1.5}B\mathcal{R}}{\gamma\rho}\exp\biggl(\frac{\ln\pi - \gamma N}{2n}\biggr) + \delta\biggr),$$

where x_* is a minimum point of g.

Remark 4 (taking account of inexact information about the value of the objective function). Both the ellipsoid method and Vaidya's method use values of the objective function to obtain the outputs (x^N) of algorithms. However, within the meaning of the statement of the saddle-point problem under consideration, it can naturally occur that the value of the objective function of the auxiliary subproblem is only available with some accuracy $\widetilde{\delta}$. In this case the above quality estimates (2.9) and (2.16) for approximate solutions produced by the methods in question must be improved by adding the term $\widetilde{\delta}$ to the right-hand sides. In fact, if $g_{\widetilde{\delta}}$ differs from g by $\widetilde{\delta}$, then for

$$\widetilde{x}^N := \operatorname*{arg\,min}_{x\in\{x_0,\dots,x_{N-1}\}} g_{\widetilde{\delta}}(x) \quad \text{and} \quad x^N := \operatorname*{arg\,min}_{x\in\{x_0,\dots,x_{N-1}\}} g(x),$$

we have $g(\widetilde{x}^N) \leqslant g(x^N) + \widetilde{\delta}$.

Remark 5. Vaidya's method with δ-subgradients can also be used in the case when information about a δ-inexact subgradient is available instead of the exact subgradient or δ-subgradients (see Lemma 3).

Remark 6 (comparing the complexity results for the ellipsoid method and Vaidya's method). Regarding the number of iterations needed to attain the prescribed accuracy (with respect to the function) of the solution of the minimization problem, the ellipsoid method if inferior to Vaidya's. In fact, the estimate for the number of iterations in the ellipsoid method depends quadratically on the dimension of the space, whereas in Vaidya's method it is proportional to $n\ln n$.

On the other hand the complexity per iteration in the ellipsoid method is less than in Vaidya's method. In fact, to perform an iteration in Vaidya's method we need to find the inverse of a square matrix of order n, while in the ellipsoid method it suffices to multiply a matrix of this size by a vector.

2.2.2. *Accelerated methods for composite optimization problems in spaces of high dimension.* In this subsection we consider the approaches to auxiliary subproblems arising in the solution of the primal problems (1.1) and (2.1) used in the case when these subproblems are of high dimension. More precisely, we describe methods for convex composite minimization problems of the form

$$\min_{y\in\mathbb{R}^m}\bigl\{U(y) := u(y) + v(y)\bigr\}, \tag{2.17}$$

where u is a μ-strongly convex function with L_u-Lipschitz gradient and v is a convex function.

Algorithm 4 (accelerated meta-algorithm (AM) for problem (2.17); see [4]).

```
Require: the number of iterations $K \geqslant 1$, the initial point $z_0$, the parameter $H > 0$.
$A_0 = 0$, $y_0 = z_0$.
for $k = 0, \dots, K-1$ do
    $\lambda_{k+1} = \frac{1}{2H}$,
    $a_{k+1} = \frac{\lambda_{k+1} + \sqrt{\lambda_{k+1}^2 + 4\lambda_{k+1}A_k}}{2}$, $A_{k+1} = A_k + a_{k+1}$,
    $\widetilde{z}_k = \frac{A_k}{A_{k+1}} y_k + \frac{a_{k+1}}{A_{k+1}} z_k$,
    $y_{k+1} = \arg\min_{y \in \mathbb{R}^d} \left\{ u(\widetilde{z}_k) + \langle \nabla u(\widetilde{z}_k), y - \widetilde{z}_k \rangle + v(y) + \frac{H}{2}\|y - \widetilde{z}_k\|_2^2 \right\}$,   (2.18)
    $z_{k+1} := z_k - a_{k+1}\nabla u(y_{k+1}) - a_{k+1}\nabla v(y_{k+1})$,
end for
Ensure: $\mathrm{AM}(z_0, K) := y_K$.
```

Algorithm 5 (restarted accelerated meta-algorithm; see [4]).

```
Require: the number of restarts $K \geqslant 1$, the initial point $z_0$, the parameters $H$ and $\mu > 0$.
for $k = 0, \dots, K-1$ do
    $N_k = \left\lceil \sqrt{\frac{32H}{\mu}} \right\rceil$,
    $z_{k+1} := \mathrm{AM}(z_k, N_k)$ (Algorithm 4),
end for
Ensure: $z_K$.
```

Theorem 5 (complexity estimate for the restarted accelerated meta-algorithm; see [4]). *Let z_N be the output of Algorithm 5 after N iterations. If $H \geqslant 2L_u$, then the total number of iterations (2.18) needed to attain accuracy $U(z_N) - U(y_*) \leqslant \varepsilon$ is*

$$N = O\left(\sqrt{\frac{H}{\mu}} \ln\left(\frac{\mu R_y^2}{\varepsilon}\right)\right), \tag{2.19}$$

where $R_y = \|y^0 - y_*\|_2$ and y_* is an exact solution of the problem (2.17).

Remark 7 (oracle complexity separation). If v has an L_v-Lipschitz gradient, then we can regard the auxiliary problem (2.18) as a smooth strongly convex problem. To solve it we can also use the restarted accelerated meta-algorithm by setting

$$u^{\mathrm{new}} := v, \qquad v^{\mathrm{new}}(y) := u(\widetilde{z}_k) + \langle \nabla u(\widetilde{z}_k), y - \widetilde{z}_k \rangle + \frac{H}{2}\|y - \widetilde{z}_k\|_2^2,$$

$$\mu^{\mathrm{new}} := \frac{H}{2} \quad \text{and} \quad H^{\mathrm{new}} := 2L_v$$

and applying the technique of restarts just like this was done in [4]. Under the condition $L_u \leqslant L_v$, this makes it possible to obtain an ε-solution of the problem (2.17) after

$$\begin{gathered} O\biggl(\sqrt{\frac{H}{\mu}}\ln\biggl(\frac{\mu R_y^2}{\varepsilon}\biggr)\biggr) \quad \text{rounds of calculation of } \nabla u \text{ and} \\ O\biggl(\sqrt{\frac{L_v}{\mu}}\ln\biggl(\frac{\mu R_y^2}{\varepsilon}\biggr)\biggr) \quad \text{rounds of calculation of } \nabla v. \end{gathered} \tag{2.20}$$

2.2.3. *Complexity estimates for algorithms for saddle-point problems using the ellipsoid or Vaidya's method for subproblems of low dimension.* To find an ε-solution of the problem (2.1) under Assumption 2 we propose to use the following approach.

Approach 1 (x is of low dimension). Algorithm 1 is applied to the problem (2.1) with parameters $\widetilde{\varepsilon} := \varepsilon/2$, $\mathcal{M}_1$ being Vaidya's method (Algorithm 3) and $\mathcal{M}_2$ being the restarted accelerated meta-algorithm (Algorithm 5).

We use the proposition from § 2.1 to establish the complexity of Approach 1. To do this, using the notation from the statement of the above proposition, we need to write out the dependencies $N_1(\varepsilon,\delta)$, $\delta(\widetilde{\varepsilon})$, and $N_2(\widetilde{\varepsilon})$. By Lemma 1 the accuracy $\widetilde{\varepsilon} = \varepsilon/2$ of the solution of the inner problem yields accuracy $\delta = \varepsilon/2$ of the δ-subgradient. As we can see from Theorem 4, the number of iterations in Vaidya's method is

$$N_1\biggl(\varepsilon, \frac{\varepsilon}{2}\biggr) = \biggl\lceil \frac{2n}{\gamma}\ln\biggl(\frac{2n^{1.5}B\mathcal{R}}{\gamma\rho\varepsilon}\biggr) + \frac{\ln\pi}{\gamma}\biggr\rceil.$$

The restarted accelerated meta-algorithm (Algorithm 5) is applied to the functions $u(\,\cdot\,) := -F(x,\,\cdot\,)$ and $v(\,\cdot\,) := h(\,\cdot\,)$ (if $L_{yy} > L_h$ in Assumption 2, b), then u and v must be interchanged). Prior to writing down its complexity, we note that the mapping $y^*(x) := \arg\max_{y\in Q_y} S(x,y)$ is continuous by virtue of the continuity of S and its strong convexity with respect to y. Therefore, the set $\{y^*(x) \mid x \in Q_x\}$ is bounded as the image of a compact set. We denote its diameter by R_y.

If h is prox-friendly (*case* 1), then, according to (2.19), an $\varepsilon/2$-solution of the inner problem can be obtained after

$$N_2\biggl(\frac{\varepsilon}{2}\biggr) = O\biggl(\sqrt{\frac{L_{yy}}{\mu_y}}\ln\biggl(\frac{\mu_y R_y^2}{\varepsilon}\biggr)\biggr) \quad \begin{array}{l}\text{rounds of calculation of } \nabla_y F \\ \text{and solution of problem (2.7).}\end{array}$$

If h has an L_h-Lipschitz gradient (*case* 2), then, according to (2.20), an $\varepsilon/2$-solution of the inner problem can be obtained after

$$\begin{gathered} N_2^F\biggl(\frac{\varepsilon}{2}\biggr) = O\biggl(\sqrt{\frac{L_{yy}}{\mu_y}}\ln\biggl(\frac{\mu_y R_y^2}{\varepsilon}\biggr)\biggr) \quad \text{rounds of calculation of } \nabla_y F \text{ and} \\ N_2^h\biggl(\frac{\varepsilon}{2}\biggr) = O\biggl(\sqrt{\frac{L_h}{\mu_y}}\ln\biggl(\frac{\mu_y R_y^2}{\varepsilon}\biggr)\biggr) \quad \text{rounds of calculation of } \nabla h. \end{gathered}$$

The above complexity estimates and the proposition in § 2.1 imply a result stated in Theorem 2. Instead of Vaidya's method, the ellipsoid method can be used as $\mathcal{M}_1$. By Theorem 3 its complexity is

$$N_1\left(\varepsilon, \frac{\varepsilon}{2}\right) = \left\lceil 2n^2 \ln\left(\frac{2B\mathcal{R}}{\rho\varepsilon}\right)\right\rceil.$$

In this case we have similar complexity estimates, namely, an ε-solution of the problem (2.1) can be obtained:

- under Assumption 2, after

$$O\left(n^2 \ln\frac{1}{\varepsilon}\right) \quad \text{rounds of calculation of } \nabla_x F \text{ and } \nabla r;$$

- under Assumption 2, a), after

$$O\left(n^2\sqrt{\frac{L_{yy}}{\mu_y}}\ln^2\frac{1}{\varepsilon}\right) \quad \begin{array}{l}\text{rounds of calculation of } \nabla_y F\\ \text{and solution of problem (2.7);}\end{array}$$

- under Assumption 2, b), after

$$O\left(n^2\sqrt{\frac{L_{yy}}{\mu_y}}\ln^2\frac{1}{\varepsilon}\right) \quad \text{rounds of calculation of } \nabla_y F \text{ and}$$

$$O\left(n^2\sqrt{\frac{L_h}{\mu_y}}\ln^2\frac{1}{\varepsilon}\right) \quad \text{rounds of calculation of } \nabla h.$$

Note that when the function h is prox-friendly (Assumption 2, a), $\mathcal{M}_2$ in Approach 1 can, for example, be the method of similar triangles (see [22]), which makes it possible to remove the requirement $Q_y \equiv \mathbb{R}^m$ from Assumption 2 while preserving the same complexity estimates.

Another interesting case arises when $Q_y = [a_1, b_1] \times \dots \times [a_m, b_m]$ is a hyperrectangle and S is separable with respect to y, that is, for $y = (y_1, y_2, \dots, y_m) \in Q_y$ we have $S(x, y) = \sum_{i=1}^m S_i(x, y_i)$, where for any $x \in Q_x$ the functions $S_i(x, y_i)$ are continuous and unimodal in y_i. Then we can remove the requirement of the smoothness and strong concavity of S in y from Assumption 2 and reduce the auxiliary problem (2.3) to m one-dimensional maximization problems

$$\max_{y_i \in [a_i, b_i]} S_i(x, y_i), \qquad i = 1, \dots, m.$$

These problems can be solved using the dichotomy method (segment bisection method) with accuracy $\varepsilon/(2m)$, which guarantees accuracy $\varepsilon/2$ of the solution of the auxiliary problem (2.3). This approach makes it possible to obtain an ε-solution of problem (2.1) after $O\left(n \ln\frac{n}{\varepsilon}\right)$ rounds of calculation of $\nabla_x F$ and $O\left(mn \ln\frac{n}{\varepsilon}\ln\frac{m}{\varepsilon}\right)$ rounds of calculation of $S(x, y)$.

Finally, we consider the case when, instead of the outer variables x, the inner ones y are of low dimension. Assume that F is convex with respect to x and μ_y-strongly concave with respect to y and also that for any $x \in Q_x$ and $y, y' \in Q_y$

$$\|\nabla_x F(x,y) - \nabla_x F(x',y)\|_2 \leqslant L_{xx}\|x - x'\|_2,$$
$$\|\nabla_x F(x,y) - \nabla_x F(x,y')\|_2 \leqslant L_{xy}\|y - y'\|_2$$

and

$$\|\nabla_y F(x,y) - \nabla_y F(x',y)\|_2 \leqslant L_{xy}\|x - x'\|_2,$$

where $L_{xx}, L_{xy} < \infty$. Also assume that the function r is μ_x-strongly convex and prox-friendly. The outer problem (2.4) (minimization of g) is multidimensional in this setting. As shown in [1], to minimize g we can use the fast gradient method with an analogue of inexact oracles, namely, the (δ, L)-model of the objective function at an arbitrary prescribed point in Q_x. Therefore, we propose the following approach to solve (2.1).

Approach 2 (y is of low dimension). The outer problem (2.4) is solved using the fast gradient method with (δ, L)-oracle for strongly convex composite optimization problems (see [1], Algorithm 4). The inner problem (2.3) is solved by Vaidya's method (Algorithm 3 with $\delta = 0$).

We can prove that such an approach makes it possible to obtain an ε-solution of problem (2.1) after

$$O\biggl(\sqrt{\frac{L}{\mu_x}}\ln\frac{1}{\varepsilon}\biggr) \quad \text{rounds of calculation of } \nabla_x F \text{ and solution of problem (1.2) and}$$

$$O\biggl(m\sqrt{\frac{L}{\mu_x}}\ln\frac{1}{\varepsilon}\ln\frac{m}{\varepsilon}\biggr) \quad \text{rounds of calculation of } \nabla_y F \text{ and } \nabla h,$$

where $L = L_{xx} + 2L_{xy}^2/\mu_y$.

2.3. Multidimensional dichotomy method for optimization problems of low dimension on a hypercube and its applications to saddle-point problems. In this subsection we consider the convex-concave saddle-point problem (one of the composite terms, r, is assumed to be identically equal to zero in problem (1.1))

$$\max_{y \in Q_y}\Bigl\{\min_{x \in Q_x} S(x,y) := F(x,y) - h(y)\Bigr\}. \tag{2.21}$$

Assume that the following holds for problem (2.21).

Assumption 3. $Q_x \subseteq \mathbb{R}^n$ is a hypercube with finite side length, $Q_y \subseteq \mathbb{R}^m$ is a nonempty convex compact set, the dimension n is very low (below 5), the function $\widehat{S}$ has the form

$$\widehat{S}(x,y) = S(x,y) = F(x,y) - h(y),$$

where the μ_y-strongly convex function h is continuous, the functional F is defined in a neighbourhood of the set $Q_x \times Q_y$, convex with respect to x and concave with respect to y. Assume that F is sufficiently smooth; more precisely, for arbitrary $x, x' \in Q_x$ and $y, y' \in Q_y$ we have

$$\begin{aligned} \|\nabla_x F(x,y) - \nabla_x F(x',y)\|_2 &\leqslant L_{xx}\|x - x'\|_2, \\ \|\nabla_x F(x,y) - \nabla_x F(x,y')\|_2 &\leqslant L_{xy}\|y - y'\|_2 \end{aligned} \tag{2.22}$$

and

$$\begin{aligned} \|\nabla_y F(x,y) - \nabla_y F(x',y)\|_2 &\leqslant L_{xy}\|x - x'\|_2, \\ \|\nabla_y F(x,y) - \nabla_y F(x,y')\|_2 &\leqslant L_{yy}\|y - y'\|_2, \end{aligned} \tag{2.23}$$

where $L_{xx}, L_{xy}, L_{yy} < +\infty$. As before (see Assumption 2), assume that one of conditions a) (*case* 1) and b) (*case* 2) holds.

For problem (2.21) we introduce the function

$$f := \max_{y \in Q_y} S(x,y).$$

We denote the diameter of the set Q_x by $R = \max_{x_1,x_2 \in Q_x} \|x_1 - x_2\|$. If the length of each side of the hypercube is a, then $R = a\sqrt{n}$.

Like in §§ 2.1 and 2.3, we consider approaches to (2.21) based on a system of auxiliary minimization problems. However, to solve low-dimensional (outer) subproblems on a hypercube we will use an analogue of the dichotomy method. Thus, first we describe this approach to the solution of convex minimization problems on a multidimensional hypercube using inexact gradients in iterations.

We consider an optimization problem of the form

$$\min_{x \in Q_x} f(x), \tag{2.24}$$

where f is a Lipschitz function with constant M_f which has a Lipschitz gradient with constant L_f, and is μ_f-strongly convex; Q_x is a finite hypercube. Below in this subsection, we describe and analyze Algorithm 6 (the multidimensional dichotomy method on a hypercube of dimension $n \geqslant 2$), which is an analogue of Nesterov's two-dimensional minimization method on a square (see [22], Exercise 4.2).

Algorithm 6 (multidimensional dichotomy method).

```
Require: the set $Q = \bigotimes_{k=1}^{n} [a_k, b_k]$, the required accuracy $\varepsilon$ in terms of the
    function, a procedure for computing an inexact gradient $\nu(\mathbf{x})$ that returns an element in the set $\{\nu(\mathbf{x}) \mid \|\nu(\mathbf{x}) - \nabla f(\mathbf{x})\|_2 \leqslant \widetilde{\delta}\}$, the initial approximation $\mathbf{x}$, the required number of iterations $N^*$.
if $Q = \{x\}$ then
  return $x$
end if
while $N \leqslant N^*$ do
  for $i = 1, \dots, n$ do
    $c := \dfrac{a_i + b_i}{2}$
    Fixing one coordinate:
        $Q_{\mathrm{new}} := \{x \in Q \mid x_i = c\}.$
```

```
      A subroutine for computing an inexact gradient in the auxiliary
      problem
          $\nu_{\text{new}}(\mathbf{x}) := \big(\nu_1(\mathbf{x}) \dots \nu_{i-1}(\mathbf{x})\nu_{i+1}(\mathbf{x}) \dots \nu_n(\mathbf{x})\big).$
      Recursive call of the multidimensional dichotomy procedure for
      the hypercube $Q_{\text{new}}$ of dimension reduced by one and with the
      new required accuracy $\widetilde{\varepsilon} = \frac{\mu_f}{128L^2R^2}\varepsilon^2$ (see (2.26))
          $\mathbf{x} := \text{Dichotomy}(Q_{\text{new}}, \widetilde{\varepsilon}, \nu_{\text{new}}).$
      $g := \nu_i(\mathbf{x})$
      if $g > 0$ then
        $Q[i] := [a_i, c]$
      else
        $Q[i] := [c, b_i]$
      end if
    end for
    $\mathbf{x} := \left(\frac{a_1 + b_1}{2} \dots \frac{a_n + b_n}{2}\right)^{\top}$
  end while
  return x
```

The following result holds for Algorithm 6.

Theorem 6. *Assume that f in (2.24) is an M_f-Lipschitz μ_f-strongly convex function that has an L_f-Lipschitz gradient. To attain accuracy ε (with respect to the function) of the output point of Algorithm 6 it suffices to perform*

$$O\left(2^{n^2} \log_2^n\left(\frac{CR}{\varepsilon}\right)\right), \quad \textit{where } C = \max\left(M_f, \frac{4(M_f + 2L_fR)}{L_f}, \frac{128L_f^2}{\mu_f}\right), \tag{2.25}$$

calls of the subroutine for computing $\nu(\mathbf{x})$, where $\nu(\mathbf{x})$ is an approximation of the gradient ∇f such that $\|\nabla f(\mathbf{x}) - \nu(\mathbf{x})\|_2 \leqslant \widetilde{\delta}(\mathbf{x})$ for any current point $\mathbf{x}$. The accuracy $\widetilde{\delta}(x)$ is derived from condition (2.28), while the accuracy of solving the auxiliary problems is specified by (2.26).

Using this result and auxiliary minimization methods described in § 2.2, we can make the following conclusions. Assume that the solution with respect to the low-dimensional variable is obtained by the multidimensional dichotomy method. Then accuracy ε in the sense of Definition 1 is attained for the saddle-point problem (2.21) after the following number of operations:

$$O\left(2^{n^2} \log_2^n\left(\frac{CR}{\varepsilon}\right)\right) \quad \text{calls of the subroutine of computing } \nabla_x S(x, y);$$

- in case 1

$$O\left(2^{n^2} \sqrt{\frac{L_{yy}}{\mu_y}} \log_2^{n+1}\left(\frac{CR}{\varepsilon}\right)\right) \quad \text{rounds of calculation of } \nabla_y F(x, y) \text{ and solution of subproblem (2.7);}$$

- in case 2

$$O\biggl(\sqrt{\frac{L_h}{\mu_y}}\log_2^{n+1}\biggl(\frac{CR}{\varepsilon}\biggr)\biggr) \quad \text{rounds of calculation of } \nabla h(y) \text{ and}$$

$$O\biggl(2^{n^2}\biggl(\sqrt{\frac{L_h}{\mu_y}}+\sqrt{\frac{L_{yy}}{\mu_y}}\biggr)\log_2^{n+1}\Bigl(\frac{CR}{\varepsilon}\Bigr)\biggr) \quad \text{rounds of calculation of } \nabla_y F(x,y).$$

2.3.1. *The description of the multidimensional dichotomy method.* The multidimensional dichotomy method involves drawing a separating hyperplane through the centre of the hypercube parallel to one of its faces and solving recursively the auxiliary optimization problem with some accuracy $\widetilde{\varepsilon}$, the choice of which is discussed below. At the point of the approximate solution an inexact gradient $\nu(x)$ is computed such that $\|\nu(x)-\nabla f(x)\|_2\leqslant\widetilde{\delta}$ for an appropriate $\widetilde{\delta}$. Then we take its component at the current point that corresponds to the fixed coordinate axis on the hyperplane in question; depending on its sign, we take the part of the hypercube not containing the inexact gradient. In one iteration this procedure is executed for each face of the hypercube. Iterations in the framework of the method are executed for the main hypercube until the size R of the remaining domain becomes less than ε/M_f, which guarantees convergence with accuracy ε with respect to the function. The stopping condition for the auxiliary subproblem guaranteeing that an acceptable accuracy of the original problem is attained will be described below in detail (Theorem 10).

We discuss the validity of Algorithm 6. Below we use the following notation. Assume that a set $Q\subset\mathbb{R}^n$ and its cross-section of the form $Q_k=\{x\in Q \mid x_k=c\}$ for some $c\in\mathbb{R}$ and $k=1,\dots,n$ are considered at the current level of recursion. Let ν be a vector in $\mathbb{R}^n$. We introduce the projections $\nu_{\|Q_k}$ and $\nu_{\perp Q_k}$ of ν onto Q_k and its orthogonal complement in the space $\mathbb{R}^n$, respectively. Note that $\nu_{\perp Q_k}$ is a scalar.

We present the following auxiliary result.

Lemma 4. *Let f be a continuously differentiable convex function. Consider the minimization problem on the set $Q_k\subset Q$ for this function. If $\mathbf{x}_*$ is a solution of this problem, then there exists a conditional subgradient $g\in\partial_Q f(\mathbf{x}_*)$ of f on Q_x such that $g_{\|}=0$.*

Note that Algorithm 6 does not converge for all convex functions even if we assume that all auxiliary one-dimensional minimization subproblems can be solved exactly. In this connection we mention an example in [16] of a nonsmooth convex function for which the multidimensional dichotomy method does not converge.

The next statement is a generalization of results (see [16]) on the convergence of Nesterov's method on a square to the multidimensional case.

Theorem 7. *Assume that f is convex and has an L_f-Lipschitz gradient for some $L_f>0$. Let $\nu(\mathbf{x})=\nabla f(\mathbf{x})$ and $\nu_{\perp Q_k}(\mathbf{x})=\bigl(\nu(\mathbf{x})\bigr)_{\perp Q_k}$ for any current point $\mathbf{x}$. If all auxiliary subproblems are solved with accuracy*

$$\widetilde{\varepsilon}\leqslant\frac{\mu_f\varepsilon^2}{128L_f^2R^2} \tag{2.26}$$

(with respect to the function), then the part of the feasible set remaining after each removal of some portion of it (in accordance with parts 11–14 *of Algorithm* 6*) contains the solution* $\mathbf{x}_*$ *of the original problem on the hypercube* Q_x*.*

This estimate requires an accuracy of order ε^{2k} (with respect to the function) of the solution of the auxiliary problem at the kth level of recursion. Thus, in view of the fact that the maximum recursion depth is $n-1$, in the worst case we need to solve the problem with accuracy ε^{2n-2} with respect to the function at each step of the algorithm.

Remark 8. The μ_f-strong convexity of f is only necessary to have a theoretical estimate for the sufficient accuracy of the solution of the auxiliary subproblems in iterations of Algorithm 6 that guarantees the required quality of the solution of problem (2.24) in linear time. To implement the multidimensional dichotomy method in practice, it is not necessary to know the constant μ_f nor to assume that $\mu_f > 0$; this was significant for setting up experiments.

It is intuitively clear that, for a function with Lipschitz gradient, a 'large' value of the orthogonal component cannot decrease significantly and therefore change its direction when we are close to the exact solution (of the auxiliary subproblem). On the other hand, when this component is small and the auxiliary problem is solved on a multidimensional parallelepiped Q_k, we can choose this point to be the required solution of the problem on Q_k. We propose an approach to choosing accuracy for auxiliary subproblems based on the above idea.

We start with a result on the necessary accuracy of the solution of auxiliary subproblems which guarantees that the required exact solution remains in the feasible set after the removal of its parts in iterations of the method. In what follows we let Δ denote the accuracy of the solution of the auxiliary subproblems (part 9 in Algorithm 6) with respect to the argument (part 10 in Algorithm 6).

Theorem 8. *Assume that* f *is convex and has an* L_f*-Lipschitz gradient for* $L_f > 0$ *and that* $\mathbf{x}$ *is an approximate solution of the auxiliary subproblem obtained at some iteration in the method. At each iteration let* $\nu(\mathbf{x}) = \nabla f(\mathbf{x})$ *and* $\nu_{\perp Q_k}(\mathbf{x}) = \big(\nu(\mathbf{x})\big)_{\perp}$*, and let the approximation* $\mathbf{x}$ *satisfy*

$$\Delta \leqslant \frac{|\nu_{\perp Q_k}(\mathbf{x})|}{L_f}.$$

Then the part of the feasible set remaining after each removal of a portion of it contains the solution $\mathbf{x}_*$ *of the original problem on the hypercube.*

The estimate in Theorem 8 can imply a very low rate of convergence of Algorithm 6 if the projection of the gradient vector at any current point onto the orthogonal complement of the subspace under consideration decreases rapidly with approach to the solution point. For this reason we state a result with an alternative stopping condition for auxiliary subproblems. We let Q_k denote the subset (line segment or part of a plane or a hyperplane) in the original hypercube Q on which the auxiliary problem is solved.

Theorem 9. *Assume that f is a convex M_f-Lipschitz function with L_f-Lipschitz gradient $(M_f, L_f > 0)$. Then to attain accuracy ε (with respect to the function) of the solution of problem (2.24) on the set Q_x it suffices that*

$$\Delta \leqslant \frac{\varepsilon - R|\nu_{\perp Q_k}(\mathbf{x})|}{L_f + M_f R}$$

for any approximate solution $\mathbf{x} \in Q_k \subset Q$, where $R = a\sqrt{n}$ is the length of the diagonal of the original hypercube Q_x.

Here ε is the accuracy with respect to the function that is required for the solution of the optimization problem on Q_k, provided that the algorithm is at the kth level of recursion, that is, it solves the auxiliary problem on $Q_k \subset Q$.

Combining the above estimates we arrive at the following theorem for $n = 2$.

Theorem 10. *Let f be a convex M_f-Lipschitz function with L_f-Lipschitz gradient $(M_f, L_f > 0)$. Assume that an inexact gradient $\nu(x)$ satisfying*

$$\|\nabla f(\mathbf{x}) - \nu(\mathbf{x})\|_2 \leqslant \widetilde{\delta}(\mathbf{x}) \tag{2.27}$$

is used at an arbitrary current point x in the implementation of the method. Then for the part of the feasible set remaining after each removal of a portion of it to contain the solution $\mathbf{x}_*$ of the original problem on Q, it suffices that

$$C_f \widetilde{\delta}(\mathbf{x}) + \Delta \leqslant \max\biggl\{\frac{|\nu_{\perp Q_k}(\mathbf{x})|}{L_f}, \frac{\varepsilon - R|\nu_{\perp Q_k}(\mathbf{x})|}{M_f + L_f R}\biggr\} \tag{2.28}$$

hold for the solution $\mathbf{x}$ of the auxiliary minimization problem for f on Q_k, where $C_f = \max\biggl(\dfrac{1}{L_f}, \dfrac{R}{M_f + L_f R}\biggr)$. In this case an approximation of the required minimum with accuracy ε is attained after at most

$$N^* := \biggl\lceil \log_2\biggl(\frac{4R(M_f + 2L_f R)}{L_f \varepsilon}\biggr)\biggr\rceil \tag{2.29}$$

iterations of Algorithm 6.

Note that the stopping criterion (2.28) applies to inner subproblems, whereas the criterion for outer subproblems is the required number of iterations (2.29).

In addition, we note that, as the diameter of the part of the feasible set remaining after the removal of hyperrectangles (in accordance with Algorithm 6) is small, the estimate for the number of iterations guaranteeing ε-accuracy with respect to the function takes the form

$$O\biggl(\log_2\biggl(\varepsilon^{-1} \max\biggl(M_f, \frac{4(M_f + 2L_f R)}{L_f}\biggr)\biggr)\biggr). \tag{2.30}$$

Thus, at each level of recursion (for the hyperrectangle Q) we solve the auxiliary problem until condition (2.28) is met. In the case when (2.28) also holds at some point for the second argument of the maximum, this point is a solution of the problem on Q. Nevertheless, if the problem on Q is in addition an auxiliary problem for some hyperrectangle Q_1 of higher dimension, then the solution of the problem at higher levels of recursion does not stop.

Now we switch to describing the theoretical results concerning estimates for the convergence rate of the method proposed for strongly convex-concave saddle-point problems of the form (2.21) with a sufficiently low dimension (up to five) of one group of variables. Recall that we regard the original problem (2.21) as a minimization problem for an auxiliary convex functional of max-type using an inexact gradient at each iteration (the accuracy of finding it is controlled via solving auxiliary minimization subproblems with respect to the other group of variables). Note that, since Q_x and Q_y are compact sets, f satisfies the Lipschitz condition in view of assumptions (2.22) and (2.23).

2.3.2. *Complexity estimates for the algorithm for saddle-point problems using the multidimensional dichotomy method in low-dimensional subproblems.* To minimize with respect to the low-dimensional variable x in problem (2.21) we use the multidimensional dichotomy method (Algorithm 6) with inexact gradient (gradient with additive noise). We describe necessary conditions on the accuracy of the gradient of the objective functional that is used at each iteration. Note that by the Demyanov-Danskin theorem (see [23])

$$\nabla f(\mathbf{x}) = \nabla_x S(\mathbf{x}, \mathbf{y}(\mathbf{x})).$$

In what follows we assume that $\nu(\mathbf{x}) = \nabla_x S(\mathbf{x}, \mathbf{y}_{\widetilde{\delta}})$, where $\mathbf{y}_{\widetilde{\delta}}$ is an approximation of $\mathbf{y}(\mathbf{x})$ such that (2.27) holds. We consider two possible *conditions for computing* $\mathbf{y}_{\widetilde{\delta}}$.

1. By virtue of the above assumptions on S,

$$\|\nu(\mathbf{x}) - \nabla f(\mathbf{x})\|_2 \leqslant L_{yx}\|\mathbf{y}(\mathbf{x}) - \mathbf{y}_\delta\|_2.$$

Thus, to determine the remaining set correctly, it suffices that the inequality (2.28) holds, where $\widetilde{\delta}(\mathbf{x}) = L_{yx}\|\mathbf{y}(\mathbf{x}) - \mathbf{y}_\delta\|_2$.

The above method assumes that the inner subproblem of the problem can be solved at a linear rate with any accuracy with respect to the argument. However, if $S(\mathbf{x}, \mathbf{y})$ is μ_y-strongly concave with respect to $\mathbf{y}$, then these conditions can be replaced by convergence with respect to the function. In this case the auxiliary problem with respect to y must be solved with any accuracy

$$\delta \leqslant \frac{\mu_y}{2L_{yx}^2}\widetilde{\delta}^2 \tag{2.31}$$

with respect to the argument

2. We can find another estimate for the necessary accuracy of the solution of the auxiliary subproblems. Note that if $y_{\widetilde{\delta}}$ is a solution of a maximization problem of the form

$$f(x) = \max_{\mathbf{y}\in Q_y} S(\mathbf{x}, \mathbf{y})$$

for fixed x with accuracy δ with respect to the function, then $\nu(\mathbf{x})$ is a δ-gradient of f at $\mathbf{x}$. By Theorem 1, if the distance of the current point x to the boundary of the feasible set Q_x is sufficiently large or, more precisely, $\rho(\mathbf{x}, \partial Q) \geqslant 2\sqrt{\delta/L_f}$, then

$$\|\nu(\mathbf{x}) - \nabla f(\mathbf{x})\|_2 \leqslant 2\sqrt{L_f\delta}.$$

In this case, to identify the remaining set correctly it suffices to have (2.28) for $\widetilde{\delta}(\mathbf{x}) = 2\sqrt{L_f \delta}$, where δ is the accuracy (with respect to the function) of the solution $y_{\widetilde{\delta}}$ of the auxiliary subproblem.

Then the auxiliary problem with respect to y must be solved with an accuracy

$$\delta \leqslant \frac{2}{L_f} \widetilde{\delta}^2 \tag{2.32}$$

with respect to the function.

Hence, at each step in the multidimensional dichotomy method we compute an inexact gradient in accordance with (2.31) or (2.32).

Now we consider the auxiliary inner minimization problem with respect to the high-dimensional variable. The strategy of solving the auxiliary minimization problem with respect to the high-dimensional variable was described in § 2.2.3. More precisely, for the auxiliary maximization subproblems with respect to high-dimensional y we use the following methods depending on the class of problems (conditions on h or S) we have chosen.

1. If h is prox-friendly (*case* 1), then we use the fast gradient method for composite optimization problems (Algorithm 4).
2. If h has L_h-Lipschitz gradient (*case* 2), it is possible to use the accelerated method with oracle complexity separation (Algorithm 2).

In these cases we obtain the following estimates for the number of calls of the corresponding auxiliary subproblems that is sufficient to attain an ε-solution of problem (2.21) in the sense of Definition 1:

- in case 1,

$$O\biggl(2^{n^2}\sqrt{\frac{L_{yy}}{\mu_y}}\log_2^{n+1}\biggl(\frac{CR}{\varepsilon}\biggr)\biggr) \quad \text{rounds of calculation of } \nabla_y F(x,y) \text{ and solution of subproblem (2.7);}$$

- in case 2,

$$O\biggl(\sqrt{\frac{L_h}{\mu_y}}\log_2^{n+1}\biggl(\frac{CR}{\varepsilon}\biggr)\biggr) \quad \text{rounds of calculation of } \nabla h(y) \text{ and}$$

$$O\biggl(2^{n^2}\biggl(\sqrt{\frac{L_h}{\mu_y}}+\sqrt{\frac{L_{yy}}{\mu_y}}\biggr)\log_2^{n+1}\biggl(\frac{CR}{\varepsilon}\biggr)\biggr) \quad \text{rounds of calculation of } \nabla_y F(x,y).$$

Note that the stopping conditions for auxiliary subproblems are (2.31) and (2.32).

We also note that if the function is separable, like in § 2.2.3, then an ε-solution of (2.21) is guaranteed attainable in r $O\biggl(n\ln\dfrac{n}{\varepsilon}\biggr)$ rounds of calculation of $\nabla_x F$ and $O\biggl(mn\ln\dfrac{n}{\varepsilon}\ln\dfrac{m}{\varepsilon}\biggr)$ rounds of calculation of $S(x,y)$ under weaker smoothness conditions.

§ 3. Results of computational experiments

3.1. Statements of the problems for which the computational efficiencies of the proposed methods are compared. As an important class of saddle-point problems, we can distinguish Lagrangian saddle-point problems associated with convex programming problems. If such a problem involves two functional constraints and Slater's condition holds, then the dual maximization problem is two-dimensional and the multidimensional dichotomy method (two-dimensional case) is quite applicable to it upon localizing the feasible domain of the dual variables. In addition, our results substantiate theoretically the linear rate of convergence in the case of a smooth strongly convex functional and a convex smooth functional constraint. The results are similar if, for example, the ellipsoid method is applied to the dual problem. However, Nesterov's method (see [16]) can work faster is some situations even in the case of nonsmooth max-type functional constraints, because there is no need to find the exact gradient of the objective function at iterations.

We consider a problem of the form

$$\max_{\lambda}\Bigl\{\phi(\lambda)=\min_{x} F(x,\lambda)\Bigr\}. \tag{3.1}$$

We seek an optimal point λ_* using the multidimensional dichotomy method, by solving the auxiliary minimization problem with respect to x at each step using the fast gradient method. This was discussed in detail in § 2.3.1.

Below we distinguish several approaches examined in this paper, using which problem (3.1) can be solved.

First, we can solve the primal problem by the ellipsoid method and the auxiliary problem by the fast gradient method (see § 2.2, case 2).

Second, we can solve the problem by the fast gradient method with (δ, L)-oracle (see [18], [24] and [25]), while solving the auxiliary problem by the ellipsoid method (Algorithm 2).

Third, we can treat the problem under consideration using the multidimensional dichotomy method, while solving the auxiliary problem by the fast gradient method (see § 2.3, case 2).

Finally, the fourth way of applying our technique does not take account of the low dimension of one of the variables but uses a variant of the fast gradient method with (δ, L)-oracle of the objective function (see [18], [24] and [25]) for the outer problem and the ordinary fast gradient method (FGM) for the inner subproblem.

It is worthy of noting that we considered strongly convex-concave saddle-point problems of the form (1.1) in § 2. Nevertheless, the strong convexity of low-dimensional optimization subproblems (to which we apply the cutting plane methods or these authors' version of the multidimensional dichotomy method) is essential only for deducing theoretical complexity estimates.

Algorithms 2, 3 and 6 can be used in practice without the assumption of strong convexity. In our experiments we managed to choose auxiliary parameters in the above methods without these assumptions and to attain the desired quality of the approximate solution without using the strong convexity of problem (1.1) with respect to the variables of low dimension (in our case, these are the dual variables in Lagrangian saddle-point problems). This explains why the experimental results presented in § 3.1 are correct despite the fact that the problems under consideration are not strongly convex/concave (but only convex/concave) with respect to one group of variables.

3.2. Lagrangian saddle-point problem associated with the quadratic optimization problem. As a specific example for comparative computations, we consider the constrained quadratic optimization problem

$$\min_{\substack{x\in Q_r\subset\mathbb{R}^n\\ g^i(x)\leqslant 0,\, i=1,\dots,m}} \biggl\{ f(x) := \frac{1}{2}\|Ax-b\|_2^2 \biggr\}, \tag{3.2}$$

where $A\in\mathbb{R}^{n\times n}$, $b\in\mathbb{R}^n$, $Q_r=\{x \mid \|x\|_2\leqslant r\}$ is a Euclidean ball, and each constraint $g^i(x)$ is linear:

$$g^i(x)={c^i}^\top x+d^i, \qquad c^i\in\mathbb{R}^n, \quad d^i\in\mathbb{R}.$$

In what follows, for it to be possible to use the optimization method on a square or a triangle, we work with two convex nonsmooth constraints g_1 and g_2 that max-aggregate the original constraints:

$$g_1(x)=\max\biggl\{ g^i(x) \biggm| i=1,\dots,\biggl\lfloor\frac{m}{2}\biggr\rfloor \biggr\}$$

and

$$g_2(x)=\max\biggl\{ g^i(x) \biggm| i=\biggl\lfloor\frac{m}{2}\biggr\rfloor+1,\dots,m \biggr\}.$$

In such a statement with two constraints the original problem (3.2) has the dual problem of the form

$$\max_{\lambda_1+\lambda_2\leqslant\Omega_\lambda}\Bigl\{ \varphi(\lambda_1,\lambda_2) := \min_{x\in Q_r}\bigl\{ f(x)+\lambda_1 g_1(x)+\lambda_2 g_2(x)\bigr\}\Bigr\},$$

where the constant Ω_λ is estimated on the basis of Slater's condition as follows:

$$\Omega_\lambda=\frac{1}{\gamma}f(\widehat{x}),$$

where $\gamma=-\max\bigl\{g_1(\widehat{x}),g_2(\widehat{x})\bigr\}>0$ and $\widehat{x}$ is an interior point in the set specified by the original constraints. Thus, along with the nonnegativeness of the dual variables, we obtain that the set on which we solve the dual problem is a right triangle with legs of length Ω_λ lying on the coordinate axes.

Let A be a sparse matrix with proportion σ of nonzero entries, with random uniformly distributed positive diagonal entries $A_{ii}\propto\mathcal{U}(0,1.1)$ and random uniformly distributed nonzero nondiagonal entries $A_{ij}\neq 0, A_{ij}\propto\mathcal{U}(0,1)$; elements of the vector b are independent uniformly distributed random variables $b_i\propto\mathcal{U}(0,0.5)$; the vector c^i and the scalar d^i, specifying the ith constraint, are also randomly generated from the uniform distribution $\mathcal{U}(0,0.1)$.

We compare the running speeds of the two-dimensional dichotomy method (in what follows, the optimization method on a square), the optimization method on an (isosceles right) triangle (which is similar to the two-dimensional dichotomy method) on the set $Q = \{x \in \mathbb{R}^2_{++} \mid x_1 + x_2 \leqslant \Omega_\lambda\}$, the ellipsoid method and the fast gradient method (FGM). We describe here a variant of the *dichotomy method on an isosceles right triangle* that is used below. Each iteration in this method is performed according to the following steps.

1. At the first iteration step, a separating line segment connecting the midpoints of one leg and the hypotenuse is drawn (Fig. 1, a). Using some one-dimensional optimization method (for example, the golden section search method) the auxiliary minimization problem is solved on this segment.
2. The gradient $\nabla f(x)$ is computed at the point x of the solution of the auxiliary problem that we have obtained; then the part of the triangle into which it points is cut off.
3. If a twice smaller triangle homothetic to the original one remains after the cutoff, then we switch to the next iteration.
4. Otherwise, one of the two parts of the remaining trapezoid into which it is partitioned by the segment connecting the midpoints of the hypotenuse and the other leg of the original triangle is similarly cut off (Fig. 1, b). If a triangle remains, then we switch to the next iteration. If a square remains, then the two-dimensional dichotomy method is used for further optimization.

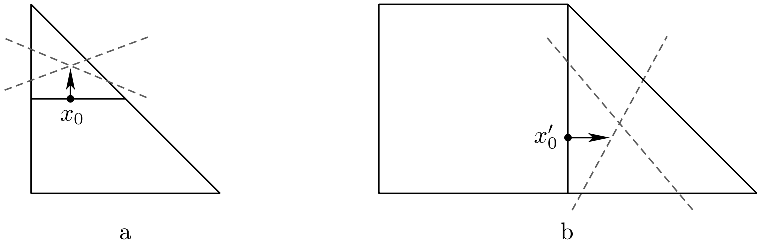

Figure 1. An illustration of the description of the dichotomy method on a triangle: (a) corresponds to the first step of the cutoff; (b) corresponds to the second step of the cutoff.

As a stopping criterion for all methods compared in this subsection, we use the condition

$$|\lambda_1 g_1(x_\delta(\lambda)) + \lambda_2 g_2(x_\delta(\lambda))| < \varepsilon,$$

where $x_\delta(\lambda)$ is a solution of the auxiliary problem $\min_{x\in Q}\{f(x)+\lambda_1 g_1(x)+\lambda_2 g_2(x)\}$ approximated with accuracy δ with respect to the function. The fulfillment of this condition guarantees that the following accuracy with respect to the function is attained for the solution of the original problem:

$$f(x_\delta(\lambda)) - \min_{\substack{x\in Q\\ g_1(x), g_2(x)\leqslant 0}} f(x) \leqslant \varepsilon + \delta.$$

At each iteration the dual factors are positive because of domain chosen for their localization (a triangle in the positive orthant), which eliminates the possibility of an untimely fulfillment of this condition at the point 0.

The auxiliary minimization problem is solved by the subgradient method (SubGM); see [26]. The number of iterations in the subgradient method is specified experimentally so that an accuracy (with respect to the function) of $\delta = 0.005$ is attained at the point of the solution of the problem (in comparison with a solution obtained after a large number of iterations of the method) and also so that the values of the constraints g_1 and g_2 at these points are nonpositive. It turns out that for $n = 400$, $r = 5$ and $\sigma = 0.005$ it suffices to perform 800 iterations using the subgradient method, while for $n = 1000$, $r = 2$ and $\sigma = 0.001$, 2500 iterations are sufficient.

As we can see from Table 2, which for different ε compares two methods for $n = 400$ and $m = 10$ and $m = 20$, the optimization method on a triangle that we propose attains the stopping criterion and thus the prescribed accuracy with respect to the function in a smaller number of iterations and a lower running time in comparison with the ellipsoid method.

Table 2. A comparison of two methods for $n = 400$

ε	m	Method on a triangle		Ellipsoid method	
		Iterations	Time, ms	Iterations	Time, ms
0.5	10	4	687	8	820
	20	4	818	8	1170
0.1	10	8	810	14	1300
	20	8	1020	14	1390
0.05	10	12	1160	16	1460
	20	14	1840	18	2140
0.01	10	22	3170	36	3530
	20	26	3260	38	3770

It can be seen from Table 3, which compares these methods for $n = 1000$ and $m = 10$, that the method on a triangle is more efficient than the ellipsoid one. The running time of the optimization method on a triangle is somewhat lower than that of the optimization method on a square, which is because the dual factors are localized in a right triangle, and therefore the lengths of the segments on which we need to solve the additional one-dimensional optimization problems at the first iteration of the methods is less. In addition, the ellipsoid method requires a much larger number of iterations and a much higher running time in comparison with the other methods in the case of a problem with m constraints (not aggregated into two constraints of max-type), due to an increase in the dimension of the problem and higher time costs for performing matrix-vector operations. As we can see from Table 3, the replacement of the subgradient method by the fast gradient method in the auxiliary high-dimensional subproblem does not change the situation.

Table 3. A comparison of three methods for $n = 1000$ and $m = 10$

ε	Method on a square		Method on a triangle		Ellipsoid method	
	Iterations	Time, s	Iterations	Time, s	Iterations	Time, s
0.5	6	6.10	6	5.41	4	6.12
0.1	12	8.92	12	8.25	16	12.7
0.05	18	12.6	16	11.2	24	23.8
0.01	24	25.3	22	24.1	30	32.5

ε	Ellipsoid method ($m = 10$, SubGM)		Ellipsoid method ($m = 10$, FGM)	
	Iterations	Time, s	Iterations	Time, s
0.5	8	15.3	6	6.27
0.1	20	26.2	10	17.6
0.05	32	38.7	34	38.8
0.01	40	49.5	40	50.2

Table 4. The work of the fast gradient method for $n = 1000$ and $m = 10$

ε	FGM		FGM ($m = 10$)	
	Iterations	Time, s	Iterations	Time, s
0.5	10	10.8	12	12.3
0.1	16	20.6	16	22.9
0.05	22	34.1	22	34.8
0.01	28	36.9	32	37.3

We compare how the method on a triangle solves the outer problem in comparison with the fast gradient method in the cases of two max-aggregated constraints and $m = 10$ original constraints (Table 4). In both variants the fast gradient method requires a larger number of iterations and a higher running time for the same accuracy than the method on a triangle.

3.3. Lagrangian saddle-point problem associated with the LogSumExp problem with linear functional constraints. We consider the LogSumExp problem with ℓ_2-regularization and the linear constraints

$$\min_{x \in \mathbb{R}^m} \left\{ \log_2 \left(1 + \sum_{k=1}^{m} e^{\alpha x_k} \right) + \frac{\mu_x}{2} \|x\|_2^2 \right\},$$

$$Bx \leqslant c, \qquad B \in \mathbb{R}^{n \times m}, \qquad c \in \mathbb{R}^n, \qquad \alpha \in \mathbb{R}.$$

We introduce the notation $\mathrm{LSE}(x) = \log_2\bigl(1 + \sum_{k=1} m e^{\alpha x_k}\bigr)$.
The Lagrangian of this problem can be expressed as

$$r(x) + F(x, y) - h(y),$$

where

$$r(x)=\frac{\mu_x}{2}\|x\|_2^2,\qquad F(x,y)=\log_2\biggl(1+\sum_{k=1}^{m}e^{\alpha x_k}\biggr)+y^\top Bx\quad\text{and}\quad h(y)=y^\top c.$$

Then the dual problem is a convex-concave saddle-point problem of the form

$$\max_{y\in\mathbb{R}^n_+}\min_{x\in\mathbb{R}^m}\bigl\{r(x)+F(x,y)-h(y)\bigr\}. \tag{3.3}$$

Note that the function $r(x)$ in the above statement of the problem is prox-friendly.

By Theorem 11 (see § 4.12),

$$y_*\in Q_y=\biggl\{y\in\mathbb{R}^n_+\biggm| y_k\leqslant\frac{f(x_0)}{\gamma}\biggr\},$$

where x_0 is an interior point in the polyhedron $Bx\leqslant c$ and $\gamma=\min_k\bigl\{c_k-[Bx]_k\bigr\}>0$. It is also straightforward to see that x must lie inside the ball $Q_x=B_{R_x}(0)$ with centre at the origin and some finite radius R_x. In fact, the value of the function at the origin is $s_0=S(0,y)=\log_2(m+1)-y^\top c$ for any $y\in Q_y$, and we can find x such that the quadratic part with respect to x is above this value for any $y\in Q_y$.

We discuss the parameters associated with the Lipschitz constants of the gradients of the functions under consideration. For r and h these obviously are

$$L_r=\mu_x\quad\text{and}\quad L_h=0.$$

It is also evident that for the function F

$$L_{xy}=\|B\|_2R_y,\qquad L_{yx}=\|B\|_2R_x\quad\text{and}\quad L_{yy}=0.$$

The constant L_{xx} (which we now compute) is the sum of the constants for the LogSumExp problem and a linear function with respect to x, which is zero. The Lipschitz constant of the gradient in the LogSumExp problem is the maximum eigenvalue of the Hessian, which is α. Thus,

$$L_{xx}=L_{\mathrm{LSE}}=\max_x\lambda_1\nabla^2\mathrm{LSE}(x)=\alpha,$$

where $\mathrm{LSE}(x)=\log_2\bigl(1+\sum_{k=1}^m e^{\alpha x_k}\bigr)$.

The parameters α_k are generated by the uniform distribution $\mathcal{U}(-\alpha_0,\alpha_0)$, $\alpha_0=0.001$. Entries of the matrix B are generated by the uniform distribution $\mathcal{U}(-k,k)$, $k=10^3$. The parameter μ_x is 0.001, and the components of the vector c are equal to 1.

We clarify the stopping condition used in performing the experiments for the methods compared. Note that if $x_\delta(\lambda)$ is a solution with accuracy δ (with respect to the function) of the problem

$$\min_{x\in Q_x}\bigl\{f(x)+\lambda^\top g(x)\bigr\}$$

and the additional condition

$$|\lambda^\top g(x_\delta(\lambda))| \leqslant \varepsilon \tag{3.4}$$

holds, then $x_\delta(\lambda)$ is an approximate solution of the minimization problem for f with accuracy $\delta + \varepsilon$ with respect to the function, that is,

$$f(x_\delta(\lambda)) - \min_{x \in Q_x} f(x) \leqslant \delta + \varepsilon.$$

In fact, we have

$$\begin{aligned} f(x_\delta(\lambda)) + \lambda^\top g(x_\delta(\lambda)) &\leqslant f(x(\lambda)) + \lambda^\top g(x(\lambda)) + \delta \\ &\leqslant \phi(\lambda_*) + \delta = f(x_*) + \lambda_*^\top g(x_*) + \delta = f(x_*). \end{aligned}$$

The last transition is due to the Karush-Kuhn-Tucker condition, which asserts that the complementary slackness condition $\lambda_i g_i(x) = 0$ for any i must be satisfied at the point $(\lambda_*, x(\lambda_*))$.

So we can choose the stopping conditions

$$\begin{cases} |\lambda^\top g(x_\delta(\lambda))| \leqslant \dfrac{\varepsilon}{2}, \\ g_i(x) \leqslant 0 \quad \forall\, i\colon \lambda_i = 0. \end{cases}$$

The first condition guarantees accuracy ε with respect to the function f, as already indicated above. The second condition is added because for $\lambda_i = 0$ the value $g_i(x)$ of noncompliance with the condition can be arbitrarily large.

Using condition (3.4) to stop the work of the method when solving (3.3), we compare the methods described above. Furthermore, we impose an additional restriction on the running time of the method. If the time limit is exceeded before the stopping condition is met, then the execution of the method is aborted and the current result is returned. This limit is set to be 100 in our problems. A dash in the corresponding tables means that the method could not stop in the allotted time for these parameters.

The Python programming language (version 3.7.3) with the installed NumPy library (version 1.18.3) was used for computations. The code was posted in a repository on the GitHub platform (see [27]).

The results of experiments for dimension $n = 2, 3, 4$ with respect to the dual variable are presented in Tables 5–7. These tables show the running time in the case when the low-dimensional problem is solved by the fast gradient method (FGM) or low-dimensional methods (such as the ellipsoid method with δ-subgradient or the multidimensional dichotomy method described in this paper), whereas the auxiliary high-dimensional problem is solved by the fast gradient method. The best (least) value in a row (for fixed ε and m) is marked bold.

The case when the low-dimensional problem is auxiliary is not included in the tables, since these methods do converge within the allocated time in all experiments. We can conclude that in the case of constraints of low dimension it is more economical to solve the low-dimensional problem as the primal one.

We discuss the results obtained. First, for all $n = 2, 3, 4$ we can see (from Tables 5, 6, and 7, respectively) that the fast gradient method yields the best

Table 5. A comparison of four methods for $n = 2$

ε	m	Running time, s			
		FGM	Ellipsoid method	Dichotomy method	Vaidya's method
10^{-3}	10^2	**0.02**	0.27	0.14	0.39
	10^3	**0.03**	0.53	0.27	0.56
	10^4	**0.45**	9.86	4.33	6.98
10^{-6}	10^2	3.48	0.45	**0.22**	0.50
	10^3	0.47	0.85	**0.45**	0.79
	10^4	**0.63**	16.72	6.16	11.10
10^{-9}	10^2	-	0.79	**0.67**	0.71
	10^3	-	1.45	**1.12**	1.24
	10^4	-	26.23	**16.09**	16.82

Table 6. A comparison of four methods for $n = 3$

ε	m	Running time, s			
		FGM	Ellipsoid method	Dichotomy method	Vaidya's method
10^{-3}	10^2	**0.05**	0.65	0.88	0.71
	10^3	**0.03**	1.30	1.56	0.91
	10^4	**0.36**	22.05	20.52	10.92
10^{-6}	10^2	2.46	1.07	-	**0.79**
	10^3	**0.53**	2.06	-	1.27
	10^4	**0.61**	37.64	-	17.66
10^{-9}	10^2	-	1.89	-	**1.17**
	10^3	-	3.63	-	**1.64**
	10^4	-	59.71	-	**25.41**

Table 7. A comparison of four methods for $n = 4$

ε	m	Running time, s			
		FGM	Ellipsoid method	Dichotomy method	Vaidya's method
10^{-3}	10^2	**0.06**	1.08	7.06	0.9
	10^3	**0.03**	2.22	12.24	1.38
	10^4	**0.37**	40.37	-	16.06
10^{-6}	10^2	3.10	1.86	-	**1.15**
	10^3	**0.56**	3.82	-	1.90
	10^4	**0.67**	-	-	25.03
10^{-9}	10^2	-	3.42	-	**2.01**
	10^3	-	6.20	-	**2.83**
	10^4	-	-	-	**33.12**

result in the case of a not very high required accuracy of $\varepsilon = 10^{-3}$. In fact, it is faster for this accuracy by at least an order of magnitude in comparison with low-dimensional methods (the ellipsoid and dichotomy methods). On the other hand, with the increase of the required accuracy the low-dimensional methods become faster that the gradient method. For example, for $n = 2$ (Table 5) we see that low-dimensional methods are faster than high-dimensional ones in the case of $\varepsilon = 10^{-9}$ for all dimensions m of the original problem.

Second, note that for $n = 2$ the proposed multidimensional dichotomy method converges faster than the ellipsoid method and Vaidya's method for all ε and m. However, the complexity of this method increases very rapidly with dimension (see the complexity estimate (2.25)), which also manifests itself in experiments. Even for $n = 3$, that is, when the dimension increases by 1, it ceases to be efficient in comparison with the other methods. For $m > 2$ and $\varepsilon = 10^{-9}$ Vaidya's method performs better than the other methods in most tests.

Third, note the nature of the dependence of the running time on the dimension m of the direct problem. For low-dimensional methods the running time grows faster with m for fixed n and ε than for high-dimensional ones. One consequence is that in our experiments for $\varepsilon = 10^{-6}$ the low-dimensional minimization methods we consider perform more efficiently in comparison with the fast gradient method only for the low dimension $m = 100$.

3.4. Lagrangian saddle-point problem associated with the LogSumExp problem with linear functional constraints and additive noise in the gradient. Under the conditions of the previous example in § 3.3 we consider a similar statement of problem (3.3) and solve it using the ellipsoid method for the outer max-problem and the fast gradient method for the inner min-problem under the additional condition that the gradient with respect to the variable y in the outer problem is obtained when the corresponding oracle in Algorithm 2 is called with some additive error. More precisely, instead of the exact gradient $\nabla_y f(y)$ of the function under the sign of the operator min, a vector v such that

$$\|\nabla_y f(y) - v\|_2 \leqslant \Delta$$

is available. According to Remark 3, the Δ-additive inexactness of the gradient can be taken into account as an additional δ-inexactness of the oracle in two ways: uniformly, using the μ-strong convexity (which is the case for the problem under consideration), or dynamically, by varying δ depending on the diameter of the current ellipsoid, which is equal (in the notation of Algorithm 2) to

$$\mathrm{diam}_k = 2 \cdot \lambda_{\max}^{1/2}(H_k),$$

where $\lambda_{\max}$ is the largest eigenvalue of the matrix.

Another inexactness of the oracle in the method for the outer problem is due to the fact that the solution of the inner problem by the fast gradient method is approximate in this case. Assume that the inner method is tuned to accuracy δ_{FGM} and executes the number of iterations that is enough to attain this accuracy according to the theoretical estimates.

Assume that the method for the outer problem, that is, the ellipsoid method, runs until the stopping condition (3.4) is met. We assume that the method is aimed at obtaining in the end an ε-solution of the general saddle-point problem. Then, using the reasoning from the last example, we need to tune the stopping condition in the ellipsoid method to accuracy $\varepsilon - \delta_{\mathrm{FGM}} - \delta$, where δ_{FGM} is the accuracy of the fast gradient method. This quantity depends on how we take account of the additive inexactness of the gradient in the same way as the actual time for meeting the stopping condition does. We examine this dependence.

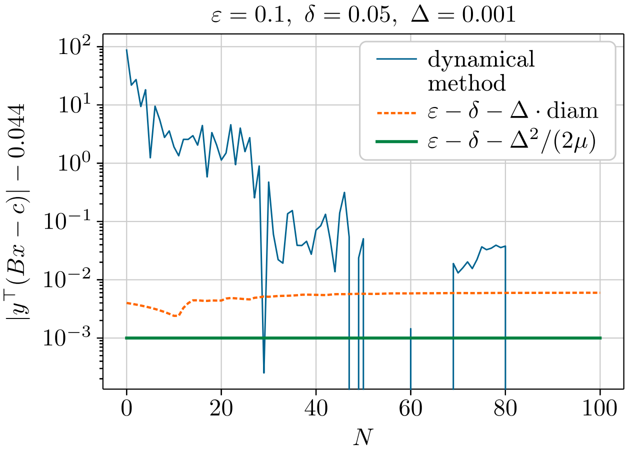

Figure 2. The actual change in the stopping condition for the ellipsoid method as applied to the Lagrangian saddle-point problem associated with the LogSumExp problem.

In Figure 2 we plot the values of the left-hand side of (3.4) and $\varepsilon - \delta_{\mathrm{FGM}} - \delta$ for two ways of taking account of the inexactness of the gradient in the particular case of the problem when $n = 3$, $m = 10$, $\varepsilon = 0.1$, $\delta = 0.05$, $\Delta = 0.001$ and $\mu = 0.0001$. As we can see, the diameters of the ellipsoids are decreasing in practice, and the bound given by the stopping condition in the case when the inexactness of the gradient is taken dynamically into account increases noticeably in comparison with the growth of the values attained at the points generated by the method. This can make it possible to reduce the number of iterations and the running time of the method until this condition is met while preserving the guarantees of the accuracy of the resulting solution.

3.5. The problem of projecting a point onto a set specified by a system of smooth constraints. Now we consider the problem of projecting a point onto a convex set with nontrivial structure specified by a system of quite a few inequality constraints with smooth functions (see [8]). The projection arises as a subproblem in many optimization algorithms that are used in their turn to solve constrained problems. It is not always possible to project precisely with reasonable algorithmic complexity. In this connection it makes sense to state the problem of finding a projection with some accuracy, that is, the problem of finding an approximate solution of a problem of the form

$$\min_{x\in\mathbb{R}^n} \|x_0 - x\|_2^2,$$
$$g_i(x) \leqslant 0, \qquad g_i \ \text{ is } L\text{-smooth} \quad \forall\, i = 1,\dots,m.$$

The Lagrangian saddle-point problem associated with this problem is as follows:

$$\max_{\lambda\in\mathbb{R}^m_+}\min_{x\in\mathbb{R}^n}\left\{\|x_0 - x\|_2^2 + \sum_{i=1}^m \lambda_i g_i(x)\right\}.$$

In the case when the number of constraints m is small an efficient approach was proposed in [8]; its algorithmic complexity depends linearly on the dimension n. This approach is based on combining the ellipsoid (or Vaidya's) method with the fast gradient method. A similar approach described in our paper has considerable distinctions, on the one hand, in its requirements for the accuracy of the solution of the auxiliary min-problem (according to the analysis proposed, it can be chosen to be $\varepsilon/2$; in the approach in [8] it is necessarily $\sim \varepsilon^4$) and, on the other hand, in the stopping condition (3.4) used (which guarantees the prescribed accuracy of the original direct problem), which can turn out to be met before the theoretically sufficient number of iterations, thus providing significant convenience for applications.

Table 8. A comparison of two methods for a system of $m = 3$ quadratic constraints

ε	Running time, s			
	$n = 200$		$n = 300$	
	Ellipsoid method	FPM	Ellipsoid method	FPM
10^{-1}	2	13	2	15
10^{-2}	3	54	12	70
10^{-4}	16	119	29	178
10^{-5}	30	171	45	271
10^{-6}	33	210	47	336

Table 8 shows the running time for the approach proposed in this paper (Algorithm 2) and the approach from [8] (Algorithm 4, Fast Projection Method — FPM) in the case of $m = 3$ constraints of the form

$$g_i(x) = (x - x_i)^\top A_i(x - x_i) - r_i \leqslant 0,$$

where the matrices A_i are positive definite and are generated randomly with entries in $\mathcal{U}(0, 0.05)$, the central points x_i have independent random components in $\mathcal{U}(-1, 1)$, and the r_i are uniformly independently randomly generated by $\mathcal{U}(0, 0.1)$. The resulting accuracy is verified in comparison with a solution obtained by one of the methods tuned to have an accuracy of $\varepsilon = 10^{-10}$ (both in terms of the direct function: $\|x_0 - x\|_2^2 \leqslant \|x_0 - x^*\|_2^2 + \varepsilon$, and in terms of the constraints: $g_i(x) \leqslant \varepsilon$ for any i). As we can see, the approach described in this paper is considerably more efficient in practice in the sense of running time because of the stopping condition used and the reduced labour expenses for the auxiliary problems.

§ 4. Proofs and used results

4.1. Proof of Lemma 1. We present a proof from [17], pp. 123–124, which uses the assumption that Q_y is compact instead of the strong concavity of S with respect to y.

Let $\nu \in \partial_x S(x, \widetilde{y})$. For any $x' \in Q_x$ we have

$$\widehat{g}(x') = \max_{y \in Q_y} S(x', y) \geqslant S(x', \widetilde{y}) \geqslant S(x, \widetilde{y}) + \langle \nu, x' - x \rangle \geqslant \widehat{g}(x) + \langle \nu, x' - x \rangle - \delta.$$

Thus, $\nu \in \partial_\delta \widehat{g}(x)$, as required.

4.2. Proof of Theorem 1. Note that the (δ, L)-subgradient $\nabla_{\delta,L} g(x)$ in the definition (2.5) corresponds to a $(2\delta, L)$-oracle of the form $(g(x) - \delta, \nabla_{\delta,L} g(x))$ in Definition 1 from [18], that is, the inequality

$$0 \leqslant g(y) - \bigl(g(x) - \delta + \langle \nabla_{\delta,L} g(x), y - x \rangle\bigr) \leqslant \frac{L}{2}\|y - x\|_2^2 + 2\delta$$

holds. It was proved in [18], § 2.2, that if $\rho(x, \partial Q_x) \geqslant 2\sqrt{\delta/L}$, then

$$\|\nabla_{\delta,L} g(x) - \nabla g(x)\| \leqslant 2\sqrt{\delta L}$$

for any subgradient $\nabla g(x)$, as required.

4.3. Proof of Lemma 2. The strong concavity of S with respect to y implies that the maximization problem (2.3) has a unique solution for any x, which we denote by $y(x)$, and that

$$S(x, y) \leqslant \underbrace{S(x, y(x))}_{\widehat{g}(x)} - \frac{\mu_y}{2}\|y - y(x)\|_2^2$$

for any $y \in Q_y$. In particular, if $\widetilde{y}$ is an $\widetilde{\varepsilon}$-solution of the inner problem (2.3), then

$$\|\widetilde{y} - y(x)\|_2^2 \leqslant \frac{2}{\mu_y}\widetilde{\varepsilon}. \tag{4.1}$$

By the Demyanov-Danskin theorem (see [23] and [28]) the function $\widehat{g}$ is differentiable at any point $x \in Q_x$ and its gradient is

$$\nabla \widehat{g}(x) = \nabla_x S(x, y(x)). \tag{4.2}$$

Using (2.6), (4.1) and (4.2) we obtain

$$\|\nabla_x S(x, \widetilde{y}) - \nabla \widehat{g}(x)\|_2 \leqslant L_{xy}\sqrt{\frac{2\widetilde{\varepsilon}}{\mu_y}},$$

as required.

4.4. Proof of Lemma 3. 1. For any $x, x' \in Q$, $\nabla g(x) \in \partial g(x)$ and a δ_1-inexact subgradient ν at x we have

$$\begin{aligned} g(x') &\geqslant g(x) + \big\langle \nabla g(x), x' - x \big\rangle \\ &= g(x) + \big\langle \nu, x' - x \big\rangle + \big\langle \nabla g(x) - \nu, x' - x \big\rangle \\ &\geqslant g(x) + \big\langle \nu, x' - x \big\rangle - \delta_1 \operatorname{diam} Q. \end{aligned}$$

Hence a δ_1-inexact subgradient ν is a δ_2-subgradient of g at x with $\delta_2 = \delta_1 \operatorname{diam} Q$.

2. For any $x, x' \in Q$, $\nabla g(x) \in \partial g(x)$ and a δ_1-inexact subgradient ν at x, we have

$$\begin{aligned} g(x') &\geqslant g(x) + \big\langle \nabla g(x), x' - x \big\rangle + \frac{\mu}{2}\|x' - x\|_2^2 \\ &= g(x) + \big\langle \nu, x' - x \big\rangle + \big\langle \nabla g(x) - \nu, x' - x \big\rangle + \frac{\mu}{2}\|x' - x\|_2^2 \\ &\geqslant g(x) + \big\langle \nu, x' - x \big\rangle - \delta_1\|x' - x\|_2 + \frac{\mu}{2}\|x' - x\|_2^2. \end{aligned}$$

In view of the fact that

$$\delta_1\|x' - x\|_2 \leqslant \frac{\mu}{2}\|x' - x\|_2^2 + \frac{\delta_1^2}{2\mu},$$

we infer the inequality

$$g(x') \geqslant g(x) + \langle \nu, x' - x \rangle - \frac{\delta_1^2}{2\mu}.$$

Therefore, a δ_1-inexact subgradient ν is a δ_2-subgradient of g at x for $\delta_2 = \delta_1^2/(2\mu)$.

4.5. Proof of Lemma 4. If $\mathbf{x}_*$ is an interior point, then the gradient with respect to the nonfixed variables is zero because $\mathbf{x}_*$ is a minimum point. In view of the fact that $\nabla f(\mathbf{x}_*) \in \partial f(\mathbf{x}_*)$, we arrive at the assertion of the lemma.

Assume that $\mathbf{x}_*$ is a boundary point. Then the set of conditional subgradients on the hypercube Q is defined by

$$\partial_Q f(\mathbf{x}) = \partial f(\mathbf{x}) + N(\mathbf{x} \mid Q),$$

where $N(\mathbf{x} \mid Q) = \big\{\mathbf{a} \mid \langle \mathbf{a}, \mathbf{y} - \mathbf{x} \rangle \leqslant 0 \ \forall\, \mathbf{y} \in Q\big\}$.

In the case when the function is differentiable we have

$$\partial f(\mathbf{x}_*) = \{\nabla f(\mathbf{x}_*)\}.$$

The fact that $\mathbf{x}_*$ is a boundary point yields that there is a nonempty set of coordinates $\{x_j\}_j$ such that $x_j = \max_{\mathbf{y} \in Q_k} y_j$ or $x_j = \min_{\mathbf{y} \in Q_k} y_j$. We introduce the notation

$$J_+ = \Big\{ j \in \mathbb{N} \ \Big|\ x_j = \max_{\mathbf{y} \in Q_k} y_j \Big\} \quad \text{and} \quad J_- = \Big\{ j \in \mathbb{N} \ \Big|\ x_j = \min_{\mathbf{y} \in Q_k} y_j \Big\}.$$

We note that any vector a such that $a_j \geqslant 0$ for all $j \in J_+$, $a_j \leqslant 0$ for all $j \in J_-$, and $a_j = 0$ otherwise lies in the normal cone.

We also note that $(\nabla f(\mathbf{x}_*))_j \leqslant 0$ for any $j \in J_+$, $(\nabla f(\mathbf{x}_*))_j \geqslant 0$ for any $j \in J_-$, and $(\nabla f(\mathbf{x}_*))_j = 0$ otherwise. To see this, if $(\nabla f(\mathbf{x}_*))_j > 0$ for some $j \in J_+$, then there exists a vector $\mathbf{x} = \mathbf{x}_* + \alpha \mathbf{e}_k \in Q$ for some $\alpha < 0$ and $e_k^j = \delta_{kj}$, where $\delta_{kj} = 1$ for $k = j$ and $\delta_{kj} = 0$ otherwise. The value of the function at this point satisfies the inequality

$$f(\mathbf{x}) = f(\mathbf{x}_*) + \alpha(\nabla f(\mathbf{x}_*))_j + o(\alpha) < f(\mathbf{x}_*)$$

for a sufficiently small α, which contradicts the fact that $\mathbf{x}_*$ is a solution.

Choosing $\mathbf{a}$ such that $\mathbf{a}_{\|} = -\bigl(\nabla f(\mathbf{x}_*)\bigr)_{\|}$, we find a subgradient from the condition

$$\mathbf{g} = \nabla f(\mathbf{x}_*) + \mathbf{a}, \qquad \mathbf{g}_{\|} = 0.$$

4.6. Proof of Theorem 3. In our method, like in the ordinary ellipsoid method, an ellipsoid is cut at each step by a plane through its centre; after this the least-volume ellipsoid containing one of the parts is considered. We can prove (see, for example, [20]) that

$$\frac{\operatorname{vol}(\mathcal{E}_{k+1})}{\operatorname{vol}(\mathcal{E}_k)} \leqslant e^{-1/(2n)} \quad \Longrightarrow \quad \operatorname{vol}(\mathcal{E}_N) \leqslant e^{-N/(2n)} \operatorname{vol}(\mathcal{B}_{\mathcal{R}}) \tag{4.3}$$

at each step. If $w_k = 0$, then

$$g(x) \geqslant g(c_k) - \delta \quad \forall\, x \in Q_x \quad \Longrightarrow \quad g(c_k) - g(x_*) \leqslant \delta$$

by the definition of a δ-subgradient, and the assumptions of the theorem are fulfilled. Furthermore, we assume that the vector w_k is zero. If $c_k \in Q_x$, then

$$(\mathcal{E}_k \setminus \mathcal{E}_{k+1}) \cap Q_x \subseteq \bigl\{x \in Q_x \colon \langle w_k, x - c_k\rangle > 0\bigr\} \subseteq \bigl\{x \in Q_x \colon g(x) > g(c_k) - \delta\bigr\} \tag{4.4}$$

due to the definition of a δ-subgradient. We consider the set $Q_x^\varepsilon := \{(1-\varepsilon)x_* + \varepsilon x,\ x \in Q_x\}$ for $\varepsilon \in [0,1]$. Note that $Q_x^\varepsilon \subseteq \mathcal{E}_0$ and

$$\operatorname{vol}(Q_x^\varepsilon) = \varepsilon^n \operatorname{vol}(Q_x) \geqslant \varepsilon^n \operatorname{vol}(\mathcal{B}_\rho) = \left(\frac{\varepsilon\rho}{\mathcal{R}}\right)^n \operatorname{vol}(\mathcal{B}_{\mathcal{R}}).$$

For $\varepsilon > e^{-N/(2n^2)}\dfrac{\mathcal{R}}{\rho}$ relation (4.3) implies the inequality $\operatorname{vol}(Q_x^\varepsilon) > \operatorname{vol}(\mathcal{E}_N)$. Therefore, there are a step $j \in \{0,\dots,N-1\}$ and a point $x_\varepsilon \in Q_x^\varepsilon$ such that $x_\varepsilon \in \mathcal{E}_j$ and $x_\varepsilon \notin \mathcal{E}_{j+1}$. If the point c_j lay outside Q_x, then we would cut off a part of $\mathcal{E}_j$ disjoint from Q_x, which would result in a contradiction with the inclusion $x_\varepsilon \in Q_x$. Hence $c_j \in Q_x$. Then using (4.4) we obtain the inequality $g(x_\varepsilon) > g(c_j) - \delta$. Since there is $x \in Q_x$ such that $x_\varepsilon = (1-\varepsilon)x_* + \varepsilon x$, we have

$$g(x_\varepsilon) \leqslant (1-\varepsilon)g(x_*) + \varepsilon g(x) \leqslant (1-\varepsilon)g(x_*) + \varepsilon\bigl((g(x_*) + B\bigr) = g(x_*) + B\varepsilon$$

because g is convex.

We deduce that

$$\begin{gathered} g(c_j) < g(x_*) + B\varepsilon + \delta \quad \forall\, \varepsilon > e^{-N/(2n^2)}\frac{\mathcal{R}}{\rho} \\ \Longrightarrow \quad g(c_j) - g(x_*) \leqslant e^{-N/(2n^2)}\frac{B\mathcal{R}}{\rho} + \delta, \end{gathered} \tag{4.5}$$

which implies (2.9). As for Corollary 1, if, in addition, g is μ-strongly convex, that is,

$$g(x) - g(x') - \langle \nabla g(x'), x - x' \rangle \geqslant \frac{\mu}{2}\|x - x'\|_2^2 \quad \forall\, x, x' \in Q_x,$$

then, substituting in $x = c_j$ and $x' = x_*$ and using $\langle \nabla g(x_*), x - x'_* \rangle$ for any $x \in Q_x$, we obtain

$$g(c_j) - g(x_*) \geqslant \frac{\mu}{2}\|c_j - x_*\|_2^2.$$

In view of (4.5), this yields the second required assertion, inequality (2.10).

4.7. Proof of Theorem 6. Assume that we solve a problem of the form

$$\min_x f(x). \tag{4.6}$$

We estimate the complexity of the multidimensional dichotomy method (that is, the number of calls of the subroutine for computing the gradient ∇f that is sufficient to attain an ε-exact solution with respect to the function).

The proof of this theorem uses the following estimate, substantiated in Theorem 10, for the number of outer iterations required to attain an acceptable quality of an approximate solution of the minimization problem for f:

$$N = \left\lceil \log_2\left(\frac{4R(M_f + 2L_f R)}{L_f \varepsilon}\right)\right\rceil.$$

Let $T(n, R, \varepsilon)$ be the number of auxiliary minimization problems for the corresponding function of dimension $n - 1$ that are sufficient to solve the problem of dimension n on a hypercube of diameter R with accuracy ε. For $n = 0$ we set $T(0, R, \varepsilon)$. Note that one iteration requires solving n auxiliary problems. In view of this fact, we obtain a recurrence formula for the primal problem:

$$T(n, R, \varepsilon) = \sum_{k=0}^{\lceil \log_2(M_f R/\varepsilon)\rceil} nT(n - 1, R \cdot 2^{-k}, \widetilde{\varepsilon})$$

and we obtain a similar expression by taking account of all necessary auxiliary subproblems:

$$T(n, R, \varepsilon) = \sum_{k=0}^{\lceil \log_2(C_1 R/\varepsilon)\rceil} nT(n - 1, R \cdot 2^{-k}, \widetilde{\varepsilon}),$$

where $\widetilde{\varepsilon}$ is specified according to (2.26) and

$$C_1 = \max\left(M_f, \frac{4(M_f + 2L_f R)}{L_f}\right).$$

Let $C_\varepsilon = 128L_f^2/\mu_f$. Using induction on n we prove the estimate

$$T(n, R, \varepsilon) \leqslant 2^{(n^2+n)/2} \log_2^n\left(\frac{CR}{\varepsilon}\right) + O\left(\log_2^n\left(\frac{CR}{\varepsilon}\right)\right), \quad \text{where } C = 2\max(C_1, C_\varepsilon). \tag{4.7}$$

In the above notation, the coefficient 2 in the expression for C makes it possible to avoid indicating or rounding-ups in what follows.

The base case of induction is obvious:

$$T(1,R,\varepsilon)=\log_2\frac{C_1}{\varepsilon}\leqslant\log_2\biggl(\frac{CR}{\varepsilon}\biggr).$$

Assume that (4.7) is valid for some dimension n. We prove that (4.7) is also valid for $n+1$:

$$\begin{aligned}T(n+1,R,\varepsilon)&=\sum_{k=0}^{\lceil\log_2(C_1R/\varepsilon)\rceil}(n+1)T(n,R\cdot 2^{-k},\widetilde{\varepsilon})\\&\leqslant(n+1)\cdot 2^{(n^2+n)/2}\sum_{k=0}^{\lceil\log_2(C_1R/\varepsilon)\rceil}\log_2^n\biggl(\frac{CC_\varepsilon R^2}{2^{2k}\varepsilon^2}\biggr)+O\biggl(\log_2^n\biggl(\frac{CR}{\varepsilon}\biggr)\biggr).\end{aligned}$$

We estimate the sum

$$\begin{aligned}\sum_{k=0}^{\lceil\log_2(C_1R/\varepsilon)\rceil}\log_2^n\biggl(\frac{CC_\varepsilon R^2}{2^{2k}\varepsilon^2}\biggr)&\leqslant\sum_{k=0}^{\lceil\log_2(CR/\varepsilon)\rceil}\log_2^n\biggl(\frac{C^2R^2}{2^{2k}\varepsilon^2}\biggr)\\&\leqslant 2^n\cdot\int_0^{\log_2(CR/\varepsilon)+1}\biggl(\log_2\biggl(\frac{CR}{\varepsilon}\biggr)-k\biggr)^n\,dk+\log_2^n\biggl(\frac{CR}{\varepsilon}\biggr)\\&=\frac{2^n}{n+1}\biggl(\log_2^{n+1}\biggl(\frac{CR}{\varepsilon}\biggr)+1\biggr)+\log_2^n\biggl(\frac{CR}{\varepsilon}\biggr).\end{aligned}$$

Therefore,

$$T(n+1,R,\varepsilon)\leqslant 2^{((n+1)^2+(n+1))/2}\log_2^{n+1}\biggl(\frac{CR}{\varepsilon}\biggr)+O\biggl(\log_2^n\biggl(\frac{CR}{\varepsilon}\biggr)\biggr),$$

which yields the required estimate (4.7). Finally we conclude that the following number of iterations of the inexact gradient $\nu(\mathbf{x})$ is sufficient to solve problem (4.6):

$$O\biggl(2^{n^2}\log_2^n\biggl(\frac{CR}{\varepsilon}\biggr)\biggr),\qquad\text{where } C=\max\biggl(M_f,\frac{4(M_f+2L_fR)}{L_f},\frac{128L_f^2}{\mu_f}\biggr).$$

Furthermore, we solve any auxiliary problem for the current level of recursion with accuracy

$$\widetilde{\delta}=\frac{\Delta}{C_f}\geqslant 2^{-N}$$

with respect to the argument (see (2.27)), where N is equal to N^* from the assertion of Theorem 10.

4.8. Proof of Theorem 7. It was proved in [16] that, given a hypercube Q with maximum distance between its points equal to R, when we need to minimize a function on it with accuracy ε, it suffices to solve the auxiliary problem in the framework of the method with accuracy

$$\Delta\leqslant\frac{\varepsilon}{8L_fR}$$

(with respect to the argument), where R is the size of the original hypercube. This theorem was proved for dimension $n = 2$ in [16]; however, it can easily be generalized to higher dimensions. If f is a μ_f-strongly convex function, then we deduce, using the above stopping condition, that it suffices to solve the auxiliary problem with accuracy

$$\widetilde{\varepsilon} \leqslant \frac{\mu_f \varepsilon^2}{128 L_f^2 R^2}$$

(with respect to the function).

4.9. Proof of Theorem 8. In what follows we will need the obvious relation

$$\forall\, a, b \in \mathbb{R} \quad |a - b| \leqslant |b| \quad \Longrightarrow \quad ab \geqslant 0. \tag{4.8}$$

Note that the set Q_k at the kth iteration is chosen correctly if the sign of the derivative with respect to a fixed variable in the solution of the auxiliary problem coincides with the sign of this derivative in the approximate solution.

Let $\nu(\mathbf{x}) = \nabla f(\mathbf{x})$. It follows from (4.8) that for the signs of $\nu_{\perp Q_k}(\mathbf{x}_*)$ and $\nu_{\perp Q_k}(\mathbf{x})$ to be the same, it suffices to have

$$\bigl|\nu_{\perp Q_k}(\mathbf{x}_*) - \nu_{\perp Q_k}(\mathbf{x})\bigr| \leqslant |\nu_{\perp Q_k}(\mathbf{x})|,$$

where $\nu_{\perp Q_k}(\mathbf{x})$ is the projection of $\nu(\mathbf{x})$ onto the orthogonal complement of the set on which the auxiliary problem is solved.

Using the Lipschitz property of the gradient of the objective functional f we obtain the assertion of the theorem.

4.10. Proof of Theorem 9. From Lemma 4 we obtain

$$\mathbf{g} \in \partial_Q f(\mathbf{x}_*) \colon \mathbf{g}_{\|} = 0.$$

By the definition of a subgradient of f at $\mathbf{x}_*$,

$$f(\mathbf{x}^*) - f(\mathbf{x}_*) \geqslant \langle \mathbf{g}, \mathbf{x}^* - \mathbf{x}_* \rangle.$$

We use the Cauchy-Bunyakovsky-Schwarz inequality and arrive at the inequality

$$f(\mathbf{x}_*) - f(\mathbf{x}^*) \leqslant \|\mathbf{g}\|_2 a\sqrt{n}.$$

On the other hand, from the Lipschitz condition for f we derive that

$$f(\mathbf{x}) - f(\mathbf{x}_*) \leqslant M_f \Delta$$

and

$$f(\mathbf{x}) - f(\mathbf{x}^*) \leqslant M_f \Delta + \|\mathbf{g}\|_2 a\sqrt{n} = M_f \Delta + |\nu_{\perp Q_k}(\mathbf{x}_*)| R$$

for any $\mathbf{x}$ in the Δ-neighbourhood of $\mathbf{x}_*$. In view of the Lipschitz property of the gradient of f,

$$f(\mathbf{x}) - f(\mathbf{x}^*) \leqslant M_f \Delta + \bigl(|\nu_{\perp Q_k}(\mathbf{x})| + L_f \Delta\bigr) R.$$

Assume that a set of diameter Δ remains in the auxiliary problem after steps 11–15 of Algorithm 6. Then to attain accuracy ε with respect to the function in the original problem, it suffices that

$$M_f\Delta + \|\mathbf{g}\|_2 a\sqrt{n} = M_f\Delta + \bigl(|f'_\perp(\mathbf{x})| + L_f\Delta\bigr)R \leqslant \varepsilon,$$
$$\Delta(M_f + L_f R) \leqslant \varepsilon - |f'_\perp(\mathbf{x})|R$$

at some point $\mathbf{x}$ in this set.

Finally, we obtain

$$\Delta \leqslant \frac{\varepsilon - R|f'_\perp(\mathbf{x})|}{M_f + L_f R}.$$

4.11. Proof of Theorem 10. Combining the estimates from Theorems 8 and 9, we conclude that to attain accuracy ε with respect to the function in the solution of the minimization problem on the hypercube Q, we must solve each auxiliary problem until the following condition on the distance between the approximate and exact solutions of this problem is satisfied:

$$\Delta \leqslant \max\biggl\{\frac{|\nu_{\perp Q_k}(\mathbf{x})|}{L_f}, \frac{\varepsilon - R|\nu_{\perp Q_k}(\mathbf{x})|}{M_f + L_f R}\biggr\}. \tag{4.9}$$

This condition holds for $\nu(\mathbf{x}) = \nabla f(\mathbf{x})$. Assume that $\nu(\mathbf{x})$ is a vector such that

$$\|\nabla f(\mathbf{x}) - \nu(\mathbf{x})\|_2 \leqslant \widetilde{\delta}(\mathbf{x}).$$

In this case it is obvious that (4.9) holds if

$$C_f\widetilde{\delta}(\mathbf{x}) + \Delta \leqslant \max\biggl\{\frac{|\nu_{\perp Q_k}(\mathbf{x})|}{L_f}, \frac{\varepsilon - R|\nu_{\perp Q_k}(\mathbf{x})|}{M_f + L_f R}\biggr\},$$

where

$$C_f = \max\biggl(\frac{1}{L_f}, \frac{R}{M_f + L_f R}\biggr).$$

We estimate the necessary number of iterations. If the auxiliary problem at the current level of recursion is solved with accuracy $\widetilde{\delta} = \dfrac{1}{C_f}\Delta$, then we obtain the condition

$$\Delta \leqslant \frac{1}{2}\max\biggl\{\frac{|\nu_{\perp Q_k}(\mathbf{x})|}{L_f}, \frac{\varepsilon - R|\nu_{\perp Q_k}(\mathbf{x})|}{M_f + L_f R}\biggr\}.$$

Assume that the absolute value of the orthogonal component of the gradient approximation $|\nu_{\perp Q_k}(\mathbf{x}_*)|$ is q. We denote its approximation with accuracy Δ with respect to the argument by $\mathbf{x}_\Delta$.

After N iterations of the multidimensional dichotomy method as applied to the auxiliary problem, using the Lipschitz property of the gradient with constant L_f we can derive the following estimate for the gradient at the point $\mathbf{x}_\Delta$:

$$q - 2L_f R \cdot 2^{-N} \leqslant |\nu_{\perp Q_k}(\mathbf{x})| \leqslant q + 2L_f R \cdot 2^{-N}.$$

This inequality takes into account the fact that the size of the set reduces by a factor of 2^N after N dichotomy iterations, that is, $\Delta \leqslant 2^{-N}R$ after N iterations.

Hence, for the condition $\Delta \leqslant \dfrac{1}{2}\dfrac{|\nu_{\perp Q_k}(\mathbf{x})|}{L_f}$ to be satisfied, it suffices that

$$R \cdot 2^{-N} \leqslant \frac{q - 2L_f R \cdot 2^{-N}}{2L_f}.$$

For the second inequality $\Delta \leqslant \dfrac{1}{2}\dfrac{\varepsilon - R|\nu_{\perp Q_k}(\mathbf{x})|}{M_f + L_f R}$ we obtain a similar condition:

$$R \cdot 2^{-N} \leqslant \frac{1}{2}\frac{\varepsilon - qR}{M_f + L_f R} - R \cdot 2^{-N}.$$

Then we arrive at the following estimate for N:

$$R \cdot 2^{-N} \leqslant \frac{1}{4}\min_{q\geqslant 0}\max\biggl(\frac{q}{L_f}, \frac{\varepsilon - qR}{M_f + L_f R}\biggr) = \frac{1}{4}\frac{L_f}{M_f + 2L_f R} \cdot \varepsilon.$$

Thus, the number of iterations in the original algorithm that is needed to attain the required accuracy in the auxiliary problems does not exceed the quantity

$$N = \biggl\lceil \log_2\biggl(\frac{4R(M_f + 2L_f R)}{L_f\varepsilon}\biggr)\biggr\rceil.$$

4.12. The statements of some known auxiliary results.

Theorem 11 (see [22], Exercise 4.1). *Consider the problem*

$$\min_{x\in\mathbb{R}^m} f(x) \quad \textit{with } g(x) \leqslant 0, \quad g\colon \mathbb{R}^m \to \mathbb{R}^n,$$

where f and the g_i are convex functions. The Lagrangian of this problem has the form

$$\phi(y) = \min_x\bigl\{f(x) + y^\top g(x)\bigr\}.$$

Assume that x_0 is a point such that $g(x_0) < 0$. Then the solution y_* of the problem $\max_y \phi(y)$ *satisfies the inequality*

$$\|y_*\|_2 \leqslant \frac{1}{\gamma}\bigl(f(x_0) - \min_x f(x)\bigr),$$

where $\gamma = \min_k\{-g_k(x_0)\}$.

§ 5. Conclusions

In this research we have obtained complexity estimates for strongly convex-concave saddle-point problems of the form

$$\min_{x\in Q_x}\max_{y\in Q_y}\bigl\{S(x,y) := r(x) + F(x,y) - h(y)\bigr\} \tag{5.1}$$

in the case when one group of variables (x or y) has a high dimension, while the other one has a low dimension (several dozens).

The first two proposed approaches to problems of this type are based on the use of cutting plane methods (the ellipsoid or Vaidya's method) for a convex minimization (concave maximization) problem for the group of variables of low dimension. We describe both variants, involving the ellipsoid method and involving Vaidya's method, since each of them has advantages of its own: Vaidya's method leads to a better estimate for the number of iterations, whereas the ellipsoid method results in a lower complexity of iterations. In the case when the outer subproblem has a low dimension, it is important to use these methods in these authors' version, replacing the ordinary subgradient by a δ-subgradient (note that a δ-inexact subgradient can be also used here and complexity estimates in this case are asymptotically the same as $\varepsilon \to 0$). It is proposed to solve the auxiliary optimization subproblems with respect to the group of variables of high dimension by using accelerated gradient methods. This scheme has made it possible to infer acceptable complexity estimates which depend on both the conditioning of the objective function and the dimension of the space (see Theorem 2 and § 2.2.3).

Note that the first approach (when x is of low dimension) can also be applied to the case when y is of low dimension by writing an analogue of problem (5.1) in the form

$$\min_{y\in Q_y} \max_{x\in Q_x} \bigl\{h(y) - F(x,y) - r(x)\bigr\}. \tag{5.2}$$

Recall that the function r is assumed to be prox-friendly, which means that we can explicitly solve the subproblem

$$\min_{x\in Q_x} \bigl\{\langle c_1, x\rangle + r(x) + c_2\|x\|_2^2\bigr\}, \qquad c_1 \in Q_x, \quad c_2 > 0. \tag{5.3}$$

Table 9 shows the number of operations needed to solve problem (5.2) with accuracy ε with respect to y (Approach 1) or to solve the similar problem (5.1) with accuracy ε with respect to x (Approach 2).

Table 9. A comparison of the numbers of operations in Approaches 1 and 2

Approach 1	Approach 2	Operation
$O\Bigl(m \ln \dfrac{m}{\varepsilon}\Bigr)$	$O\biggl(m\sqrt{\dfrac{L_{xx}}{\mu_x} + \dfrac{2L_{xy}^2}{\mu_x\mu_y}} \ln \dfrac{m}{\varepsilon} \ln \dfrac{1}{\varepsilon}\biggr)$	rounds of calculation of $\nabla_y F$ and ∇h
$O\biggl(m\sqrt{\dfrac{L_{xx}}{\mu_x}} \ln \dfrac{m}{\varepsilon} \ln \dfrac{1}{\varepsilon}\biggr)$	$O\biggl(\sqrt{\dfrac{L_{xx}}{\mu_x} + \dfrac{2L_{xy}^2}{\mu_x\mu_y}} \ln \dfrac{1}{\varepsilon}\biggr)$	rounds of calculation of $\nabla_x F$ and solution of problem (5.3)

According to Table 9, the second approach loses to the first approach in most cases. Nevertheless, in the case when

$$m \ln m \gg \sqrt{\frac{L_{xx}}{\mu_x} + \frac{2L_{xy}^2}{\mu_x\mu_y}}$$

and the computation of $\nabla_x F$ and solution of problem (5.3) are laborious, the second approach can turn out to be more efficient.

In addition to cutting plane methods with inexact δ-subgradient, this paper also proposes an analogue of the dichotomy method for the solution of low-dimensional convex optimization problems using inexact gradients in iterations. This is called the *multidimensional dichotomy method.* In fact, it is a generalization of the ordinary (one-dimensional) dichotomy method to minimization problems for functions of n variables. It has turned out that using this approach in very low-dimensional problems is quite reasonable. Conditions for the solution of the auxiliary problem were presented, and the convergence of the method is proved when these conditions hold at each step. In addition, an estimate is obtained for the number of iterations that is sufficient for the required accuracy with respect to the function (Theorem 6). This estimate depends on the dimension of the space (this dependence is comparable with $O(2^{n^2})$) and also on the required accuracy of the solution (this dependence has the form $O(\log_2^n(1/\varepsilon))$). This result looks much worse than the estimates for the ellipsoid method with δ-subgradients described above. However, our computational experiments showed that the proposed multidimensional dichotomy method can perform more efficiently than the ellipsoid method for $n = 2$, which corresponds to the case of two constraints in the direct problem.

Based on experiments, we have compared the dichotomy method, the fast gradient method with (δ, L)-oracle, and the ellipsoid of Vaidya's method using δ-subgradients on saddle-point problems with low dimension of one of the variables. More precisely, we have performed experiments for the dual problem of the LogSumExp problem of minimizing a function with ℓ_2-regularization in dimension m and with n linear constraints. As a result of experiments, we have established the following.

First, low-dimensional methods are faster than the fast gradient method in the case of a high required accuracy. Under our conditions, this is an accuracy of $\varepsilon = 10^{-9}$.

Second, the multidimensional dichotomy method is faster than the ellipsoid method for $n = 2$. However, its running time grows critically with dimension, and its efficiently decreases significantly even for $n = 3$.

Third, it has been established that the running time of the fast gradient method for this problem does not grow as strongly with m as in the case of the ellipsoid method or the multidimensional dichotomy method.

In addition, we have compared the multidimensional dichotomy method, the cutting plane methods considered in this paper (the ellipsoid method and Vaidya's method) and the fast gradient method on the problem of the minimization of a quadratic function (for $n = 400$ and $n = 1000$) with two nonsmooth constraints max-aggregating several ($m = 10, 20$) linear constraints. For this problem the running time of the dichotomy method and its variants (the method on a triangle) has turned out to be less than that of the ellipsoid method (both with the original and aggregated constraints) and fast gradient method. We have compared different ways of taking account of inexactnesses occurring in the presence of additive noise in the values of the gradient in the case when the ellipsoid method is applied to the low-dimensional problem. When the inexactness varies with the diameter of the current ellipsoid, the method in question meets the stopping condition more rapidly than in the case of a constant inexactness estimate. Different methods have also been compared for the problem of projecting a point onto a set specified by a system of smooth functional constraints (see [8]). Our approach of applying the ellipsoid method to subproblems of low dimension while using the proposed stopping condition turns out to be more efficient in comparison with the algorithm in [8] for a similar problem. In the framework of the above computational experiments, we considered problem statements when the low-dimensional problem (in this case, with respect to the dual variables of the Lagrangian saddle-point problem) was not strongly convex (concave). Despite the fact that this paper gives a theoretical analysis of estimates for the running speeds of the methods in question only for strongly convex-concave problems, tuning the methods adequately makes it possible in practice to apply the schemes proposed to merely convex (or concave) low-dimensional subproblems with the same success and with guarantees of attaining the required quality of the solution of the problem. This is explained by the lack of necessity to assume the strong convexity (concavity) of the objective function (it is only of importance for theoretical estimates) to implement all methods applied to low-dimensional subproblems in this paper.

Bibliography

[1] M. S. Alkousa, A. V. Gasnikov, D. M. Dvinskikh, D. A. Kovalev and F. S. Stonyakin, "Accelerated methods for saddle-point problem", *Zh. Vychisl. Mat. Mat. Fiz.* **60**:11 (2020), 1843–1866; English transl. in *Comput. Math. Math. Phys.* **60**:11 (2020), 1787–1809.

[2] W. Azizian, I. Mitliagkas, S. Lacoste-Julien and G. Gidel, "A tight and unified analysis of gradient-based methods for a whole spectrum of differentiable games", *Proceedings of the twenty third international conference on artificial intelligence and statistics*, Proceedings of Machine Learning Research (PMLR), vol. 108, 2020, pp. 2863–2873, https://proceedings.mlr.press/v108/azizian20b.html.

[3] A. V. Gasnikov, P. E. Dvurechensky and Yu. E. Nesterov, "Stochastic gradient methods with inexact oracle", *Tr. Mosk. Fiz. Tekhn. Inst.* **8**:1 (2016), 41–91. (Russian)

[4] A. V. Gasnikov, D. M. Dvinskikh, P. E. Dvurechensky, D. I. Kamzolov, V. V. Matyukhin, D. A. Pasechnyuk, N. K. Tupitsa and A. V. Chernov, "Accelerated meta-algorithm for convex optimization problems", *Zh. Vychisl. Mat. Mat. Fiz.* **61**:1 (2021), 20–31; English transl. in *Comput. Math. Math. Phys.* **61**:1 (2021), 17–28.

[5] Le Thi Khanh Hien, Renbo Zhao and W. B. Haskell, *An inexact primal-dual framework for large-scale non-bilinear saddle point problem*, arXiv: 1711.03669.

[6] Guanghui Lan, *First-order and stochastic optimization methods for machine learning*, Springer Ser. Data Sci., Springer, Cham 2020, xiii+582 pp.

[7] Tianyi Lin, Chi Jin and M. I. Jordan, "Near-optimal algorithms for minimax optimization", *Proceedings of thirty third conference on learning theory*, Proceedings of Machine Learning Research (PMLR), vol. 125, 2020, pp. 2738–2779, https://proceedings.mlr.press/v125/lin20a.html.

[8] I. Usmanova, M. Kamgarpour, A. Krause and K. Levy, "Fast projection onto convex smooth constraints", *Proceedings of the* 38*th international conference on machine learning*, Proceedings of Machine Learning Research (PMLR), vol. 139, 2021, pp. 10476–10486, http://proceedings.mlr.press/v139/usmanova21a.html.

[9] A. V. Gasnikov, *Modern numerical methods of optimization. The method of universal gradient descent*, 2ns revised ed., Moscow Center for Continuous Mathematical Education, Moscow 2021, 272 pp. (Russian)

[10] B. Cox, A. Juditsky and A. Nemirovski, "Decomposition techniques for bilinear saddle point problems and variational inequalities with affine monotone operators", *J. Optim. Theory Appl.* **172**:2 (2017), 402–435.

[11] Yu. Nesterov, "Excessive gap technique in nonsmooth convex minimization", *SIAM J. Optim.* **16**:1 (2005), 235–249.

[12] Junyu Zhang, Mingyi Hong and Shuzhong Zhang, "On lower iteration complexity bounds for the convex concave saddle point problems", *Math. Program.* **194**:1–2, Ser. A (2022), 901–935.

[13] Yuanhao Wang and Jian Li, "Improved algorithms for convex-concave minimax optimization", *NIPS* 2020, Adv. Neural Inf. Process. Syst., vol. 33, MIT Press, Cambridge, MA 2020, 11 pp., http://proceedings.neurips.cc/paper/2020; arXiv: 2006.06359.

[14] A. Nemirovski, S. Onn and U. G. Rothblum, "Accuracy certificates for computational problems with convex structure", *Math. Oper. Res.* **35**:1 (2010), 52–78.

[15] A. Nemirovski, *Information-based complexity of convex programming*, Lecture notes, 1995, 268 pp., https://www2.isye.gatech.edu/~nemirovs/Lec_EMCO.pdf.

[16] D. A. Pasechnyuk and F. S. Stonyakin, "One method for minimization a convex Lipschitz-continuous function of two variables on a fixed square", *Komp'yuter. Issled. Modelirovanie* **11**:3 (2019), 379–395. (Russian)

[17] B. T. Polyak, *Introduction to optimization*, Nauka, Moscow 1983, 384 pp.; English transl., Transl. Ser. Math. Eng., Optimization Software, Inc., Publications Division, New York 1987, xxvii+438 pp.

[18] O. Devolder, F. Glineur and Yu. Nesterov, "First-order methods of smooth convex optimization with inexact oracle", *Math. Program.* **146**:1–2, Ser. A (2014), 37–75.

[19] P. M. Vaidya, "A new algorithm for minimizing convex functions over convex sets", *Math. Program.* **73**:3, Ser. A (1996), 291–341.

[20] S. Bubeck, "Convex optimization: algorithms and complexity", *Found. Trends Mach. Learn.* **8**:3–4 (2015), 231–357.

[21] E. Gladin, A. Sadiev, A. Gasnikov, P. Dvurechensky, A. Beznosikov and M. Alkousa, "Solving smooth min-min and min-max problems by mixed oracle algorithms", *MOTOR* 2021: *Mathematical optimization theory and operations research – recent trends*, Commun. Comput. Inf. Sci., vol. 1476, Springer, Cham 2021, pp. 19–40.

[22] A. V. Gasnikov and Yu. E. Nesterov, "Universal method for stochastic composite optimization problems", *Zh. Vychisl. Mat. Mat. Fiz.* **58**:1 (2018), 52–69; English transl. in *Comput. Math. Math. Phys.* **58**:1 (2018), 48–64.

[23] J. M. Danskin, "The theory of Max-Min, with applications", *SIAM J. Appl. Math.* **14**:4 (1966), 641–664.

[24] O. Devolder, *Exactness, inexactness and stochasticity in first-order methods for large-scale convex optimization*, PhD thesis, UCL, CORE, Louvain-la-Neuve 2013, vii+309 pp., https://dial.uclouvain.be/pr/boreal/en/object/boreal%3A128257/datastream/PDF_01/view.

[25] O. Devolder, F. Glineur and Yu. Nesterov, *First-order methods with inexact oracle: the strongly convex case*, CORE Discussion Papers, no. 2013/16, 2013, 35 pp., http://www.uclouvain.be/cps/ucl/doc/core/documents/coredp2013_16web.pdf.

[26] N. Z. Shor, *Minimization methods for non-differentiable functions*, Naukova dumka, Kiev 1979, 199 pp.; English transl., Springer Ser. Comput. Math., vol. 3, Springer-Verlag, Berlin 1985, viii+162 pp.

[27] *Source code for experiments on GitHub*, https://github.com/ASEDOS999/SPP.

[28] P. Bernhard and A. Rapaport, "On a theorem of Danskin with an application to a theorem of von Neumann-Sion", *Nonlinear Anal.* **24**:8 (1995), 1163–1181.

Received 26/NOV/21 and 24/OCT/22
Translated by N. BERESTOVA

Mohammad S. Alkousa
Moscow Institute of Physics and Technology,
Dolgoprudny, Moscow Region, Russia
E-mail: mohammad.alkousa@phystech.edu

Alexander V. Gasnikov
Moscow Institute of Physics and Technology,
Dolgoprudny, Moscow Region, Russia;
Research Center for Trusted Artificial Intelligence,
Ivannikov Institute for System Programming
of the Russian Academy of Science,
Moscow, Russia;
Institute for Information Transmission Problems
of the Russian Academy of Sciences
(Kharkevich Institute), Moscow, Russia;
Caucasus Mathematical Center,
Adyghe State University,
Maikop, Russia
E-mail: gasnikov.av@mipt.ru

Egor L. Gladin
Humboldt-Universität zu Berlin,
Berlin, Germany
E-mail: egor.gladin@student.hu-berlin.de

Ilya A. Kuruzov
Moscow Institute of Physics and Technology,
Dolgoprudny, Moscow Region, Russia
E-mail: kuruzov.ia@phystech.edu

Dmitry A. Pasechnyuk
Moscow Institute of Physics and Technology,
Dolgoprudny, Moscow Region, Russia
E-mail: pasechnyuk2004@gmail.com

Fedor S. Stonyakin
Moscow Institute of Physics and Technology,
Dolgoprudny, Moscow Region, Russia;
V. I. Vernadsky Crimean Federal University,
Simferopol, Russia
E-mail: fedyor@mail.ru