Adaptive Mirror Descent for the Network Utility Maximization Problem [⋆]

Anastasiya Ivanova [*] **Fedor Stonyakin** [**]
Dmitry Pasechnyuk [***] **Evgeniya Vorontsova** [****]
Alexander Gasnikov [†]

[*] *Moscow Institute of Physics and Technology, Moscow, Russia, National Research University Higher School of Economics, Moscow, Russia, Sirius University of Science and Technology, Sochi, Russia (anastasiya.s.ivanova@phystech.edu)*
[**] *Moscow Institute of Physics and Technology, Moscow, Russia, V. I. Vernadsky Crimean Federal University, Simferopol, Russia*
[***] *Saint Petersburg Lyceum 239, Russia*
[****] *Catholic University of Louvain (UCL), Belgium, Moscow Institute of Physics and Technology, Moscow, Russia*
[†] *Moscow Institute of Physics and Technology, Moscow, Russia, Institute for Information Transmission Problems, Moscow, Russia, Caucasus Mathematical Center, Adyghe State University, Russia*

Abstract: Network utility maximization is the most important problem in network traffic management. Given the growth of modern communication networks, we consider utility maximization problem in a network with a large number of connections (links) that are used by a huge number of users. To solve this problem an adaptive mirror descent algorithm for many constraints is proposed. The key feature of the algorithm is that it has a dimension-free convergence rate. The convergence of the proposed scheme is proved theoretically. The theoretical analysis is verified with numerical simulations. We compare the algorithm with another approach, using the ellipsoid method (EM) for the dual problem. Numerical experiments showed that the performance of the proposed algorithm against EM is significantly better in large networks and when very high solution accuracy is not required. Our approach can be used in many network design paradigms, in particular, in software-defined networks.

Keywords: Optimization problems, Duality, Resource allocation, Adaptive algorithms, Communication Networks, Bandwidth allocation, Utility functions

1. INTRODUCTION

One of the most significant design features of modern communication network systems is the capacity to adjust the distribution of bandwidth and other network properties for achieving best performance and reliability in real-time. The tasks of finding the best configuration and/or design parameters for networks are actually reduced to solving complex optimization problems with thousands and millions of variables.

The key question we address in this article is how the available bandwidth on the network should be distributed among competing connections. In this case, users can control the use of available bandwidth by adjusting the connection price. As mentioned earlier, in view of the huge size of such networks it is very important to develop algorithms dimension-free on the size of the system.

Thus, network utility maximization (NUM) problems in computer networks with a large number of connections are considered. Connections are used for their own purposes by consumers (users), the number of whom can also be very large. The purpose of the work is to determine the mechanism of resource allocation, which in the context of this task are available bandwidth connections. At the same time, it is necessary to ensure stable operation of the system and prevent overloads. As an optimality criterion, the sum of the utilities of all users of the network is used.

The original resource allocation framework, reduced to the maximization of aggregate concave utility functions subject to link capacity constraints, was pioneered by Kelly et al. (1998). During the last two decades NUM framework has found wide-ranging applications to wireless and sensor networks, and many other fields Palomar and Chiang (2006); Dehghan et al. (2016), for a survey, see Shakkottai and Srikant (2008) and references therein.

[⋆] The research of F. Stonyakin in Section 3 was supported by Russian Science Foundation project 18-71-00048. The research of E. Vorontsova was supported by Russian Foundation for Basic Research project 18-29-03071 mk. The research of A. Ivanova in Sections 4 and 5 was supported by Russian Foundation for Basic Research project 19-31-51001. The research of A. Gasnikov in Section 4 was funded by the Ministry of Science and Higher Education of the Russian Federation (Goszadaniye in MIPT), number 075-00337-20-03. The research of A. Gasnikov in Section 5 was partially supported by Yahoo! Research Faculty Engagement Program.

Copyright lies with the authors

Also, the mechanisms of decentralized resource allocation proposed in a monograph by Arrow and Hurwicz (1958) have since attracted much attention in economic research, see, for example, Kakhbod (2013); Campbell et al. (1987); Friedman and Oren (1995). Furthermore, the problem of resource distribution in computer networks was investigated in a recent paper by Rokhlin (2019) as well. In this paper, following Nesterov and Shikhman (2018); Ivanova et al. (2019, 2020), we additionally consider a price adjustment mechanism. The proposed approach has a practical value due to decentralization, which means that to set and adjust the price of an individual connection, only the reaction of the users who use that connection is necessary, not the reaction of all users.

2. PROBLEM STATEMENT

Consider a communication network with m connections (links) and n users (or nodes). Users exchange packets through a fixed connections' set. The network structure is given by the binary routing matrix $C = (C_i^j) \in \mathbb{R}^{m\times n}$. The columns of the matrix $\mathbf{C}_i \neq 0,\ i = 1, \ldots, n$ are boolean m-dimensional vectors such that $C_i^j = 1$ if node i is used in the connection j, otherwise $C_i^j = 0$. Capacity constraints are given by the vector $\mathbf{b} \in \mathbb{R}^m$ with strictly positive components. These constraints imply that no connection is overloaded. Users evaluate the quality of the network with utility functions $u_k(x_k)$, $k = 1, \ldots, n$, where $x_k \in \mathbb{R}_+$ is the network rate for the k–th user.

The problem of maximizing the total utility of the network under the given constraints is formulated as follows:

$$\max_{\left\{C\mathbf{x}=\sum\limits_{k=1}^{n}\mathbf{C}_k x_k\right\}\leq \mathbf{b}} \left\{ U(\mathbf{x}) = \sum_{k=1}^{n} u_k(x_k) \right\}, \tag{1}$$

where $u_k(x_k)\ k = 1, \ldots, n$ are concave functions and $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n_+$. Optimal resource allocation $\mathbf{x}^*$ is a solution of the problem (1).

Denote $g_j(\mathbf{x}) = \langle \mathbf{C}_j, \mathbf{x}\rangle - b_j,\ \ j = 1, \ldots, m$. Note that

$$|g_j(\mathbf{x}^1) - g_j(\mathbf{x}^2)| \leq \|\mathbf{C}_j\|_2 \|\mathbf{x}^1 - \mathbf{x}^2\|_2 \leq m \cdot \|\mathbf{x}^1 - \mathbf{x}^2\|_2.$$

The last inequality holds due to the definition of matrix C. So, each $g_j(\mathbf{x}),\ j = 1, \ldots, m$ is $M_g = m$-Lipschitz continuous.

For convenience, we will further consider the minimization problem equivalent to (1)

$$\min_{g_j(\mathbf{x})\leq 0,\, j=\overline{1,\, m}} f(\mathbf{x}), \tag{2}$$

where $f(\mathbf{x}) = -U(\mathbf{x})$.

3. MIRROR DESCENT FOR MANY CONSTRAINTS

Let us consider the max-type functional constraint: $g(x) = \max\limits_{j\in\overline{1,m}} g_j(x)$, which keeps the Lipschitz property and non-smoothness, provided that all functions $g_j(x),\ j = 1, \ldots, m$, satisfy these properties.

3.1 First algorithm

The first variant of Mirror Descent Algorithm for many constraints is suggested, see Algorithm 1.

Algorithm 1 Mirror Descent Algorithm for many constraints

```
Require: $\varepsilon > 0, \Theta_0 :\ d(\mathbf{x}^*) = \frac{1}{2}\|\mathbf{x}^0 - \mathbf{x}^*\|_2 \leqslant \Theta_0^2$, initial
    point $\mathbf{x}^0 = 0$
$I =: \emptyset$
$N \leftarrow 0$
repeat
    if $g_j(\mathbf{x}^N) \leqslant \varepsilon\|\nabla g_j(\mathbf{x}^N)\|_2,\ \forall\, j = \overline{1,\, m}$ then
        $\mathbf{x}^{N+1} = \left[\mathbf{x}^N - \frac{\varepsilon\nabla f(\mathbf{x}^N)}{\|\nabla f(\mathbf{x}^N)\|_2}\right]_+$
        $//\ h_N = \frac{\varepsilon}{\|\nabla f(x^N)\|_2}$
        $N \rightarrow I$
    else
        $(g_{j_N}(\mathbf{x}^N) > \varepsilon\|\nabla g_{j_N}(\mathbf{x}^N)\|_2),\ \ j_N = \overline{1,\, m})$
        $\mathbf{x}^{N+1} = \left[\mathbf{x}^N - \frac{\varepsilon\nabla g_{j_N}(\mathbf{x}^N)}{\|\nabla g_{j_N}(\mathbf{x}^N)\|_2}\right]_+$
        $//\ h_N = \frac{\varepsilon}{\|\nabla g_{j_N}(\mathbf{x}^N)\|_2}$
    end if
    $N \leftarrow N + 1$
until $2\frac{\Theta_0^2}{\varepsilon^2} \leqslant N$
Ensure: $\bar{\mathbf{x}}^N := \operatorname*{argmin}\limits_{\mathbf{x}^k,\ k\in I} f(\mathbf{x}^k)$
```

Now we will estimate the rate of convergence of the proposed method. For this we need the following auxiliary assumption (Nesterov (2018), Lemma 3.2.1). Recall that $\mathbf{x}^*$ is the solution of the problem (2).

Lemma 1. Let us define the following function:

$$\omega(\tau) = \max_{\mathbf{x}\in\mathbb{R}^n_+}\{f(\mathbf{x}) - f(\mathbf{x}^*) : \|\mathbf{x} - \mathbf{x}^*\| \leqslant \tau\}, \tag{3}$$

where τ is a positive number. Then for any $\mathbf{y} \in X$

$$f(\mathbf{y}) - f(\mathbf{x}^*) \leqslant \omega(v_f(\mathbf{y}, \mathbf{x}^*)), \tag{4}$$

where

$$v_f(\mathbf{y}, \mathbf{x}^*) = \left\langle \frac{\nabla f(\mathbf{y})}{\|\nabla f(\mathbf{y})\|}, \mathbf{y} - \mathbf{x}^* \right\rangle \text{ for } \nabla f(\mathbf{y}) \neq 0$$

and $v_f(\mathbf{y}, \mathbf{x}^*) = 0$ for $\nabla f(\mathbf{y}) = 0$.

Theorem 2. Let $\varepsilon > 0$ be a fixed number and the stopping criterion of Algorithm 1 be satisfied. Then

$$\min_{k\in I} v_f(\mathbf{x}^k, \mathbf{x}^*) \leqslant \varepsilon, \quad \max_{k\in I} g_j(\mathbf{x}^k) \leq \varepsilon M_g,\ \ j \in \overline{1, m}. \tag{5}$$

Proof.

1) If $k \in I$ (for productive steps),

$$\begin{aligned} &h_k(f(\mathbf{x}^k) - f(\mathbf{x}^*)) \leq h_k\langle\nabla f(\mathbf{x}^k), \mathbf{x}^k - \mathbf{x}^*\rangle = \varepsilon v_f(\mathbf{x}^k, \mathbf{x}^*) \\ &\leqslant \frac{h_k^2}{2}||\nabla f(\mathbf{x}^k)||_2^2 + \frac{1}{2}\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 - \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_2^2 \\ &= \frac{\varepsilon^2}{2} + \frac{1}{2}\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 - \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_2^2. \end{aligned} \tag{6}$$

2) If $k \notin I$, then $\frac{g_{j_k}(\mathbf{x}^k) - g_{j_k}(\mathbf{x}^*)}{||\nabla g_{j_k}(\mathbf{x}^k)||_2} \geq \frac{g_{j_k}(\mathbf{x}^k)}{||\nabla g_{j_k}(\mathbf{x}^k)||_2} > \varepsilon$. Therefore, the following inequalities hold

$$\begin{aligned}&\varepsilon^2 < h_k(g_{j_k}(\mathbf{x}^k) - g_{j_k}(\mathbf{x}^*)) \leqslant \frac{h_k^2}{2}||\nabla g_{j_k}(\mathbf{x}^k)||_2^2\\ &+ \frac{1}{2}\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 - \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_2^2\\ &= \frac{\varepsilon^2}{2} + \frac{1}{2}\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 - \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_2^2,\\ &\text{or} \quad \frac{\varepsilon^2}{2} < \frac{1}{2}\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 - \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_2^2. \qquad (7)\end{aligned}$$

3) After summing the inequalities (6) and (7) we have:

$$\begin{aligned}&\sum_{k\in I}\varepsilon v_f(\mathbf{x}^k, \mathbf{x}^*) \leqslant |I|\frac{\varepsilon^2}{2} - \frac{\varepsilon^2|J|}{2}\\ &+ \frac{1}{2}\|\mathbf{x}^0 - \mathbf{x}^*\|_2^2 - \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_2^2 = \varepsilon^2|I| - \frac{\varepsilon^2 N}{2} + \Theta_0^2.\end{aligned}$$

And since $\sum_{k\in I} v_f(\mathbf{x}^k, \mathbf{x}^*) \geq |I| \min\limits_{k\in I} v_f(\mathbf{x}^k, \mathbf{x}^*)$, after the stopping criterion of the algorithm holds we have

$$|I|\min_{k\in I}\varepsilon v_f(\mathbf{x}^k, \mathbf{x}^*) \leqslant \varepsilon^2|I| - \frac{\varepsilon^2 N}{2} + \Theta_0^2 \leq \varepsilon^2|I|.$$

So, $\min\limits_{k\in I} v_f(\mathbf{x}^k, \mathbf{x}^*) \leqslant \varepsilon$.

Further, for each $k \in I$ $g_j(\mathbf{x}^k) \leqslant \varepsilon||\nabla g_j(\mathbf{x}^k)||_2 \leqslant \varepsilon M_g$, $j = 1, \ldots, m$.

Now we have to show that the set of productive steps I is non-empty. If $I = \emptyset$, then $|J| = N$ and the Lipschitz continuous of g means that $N \geqslant \frac{2\Theta_0^2}{\varepsilon^2}$. On the other hand, from (7) we have:

$$\frac{\varepsilon^2 N}{2} < \frac{1}{2}\|\mathbf{x}^0 - \mathbf{x}^*\|_2^2 \leqslant \Theta_0^2,$$

which leads us to the controversy, so $I \neq \emptyset$.

Now let us show how to estimate the quality of the solution by the function basing on the previous theorem for Lipschitz continuous function.

Corollary 3. Let f satisfy the Lipschitz condition

$$|f(\mathbf{x}) - f(\mathbf{y})| \leqslant M_f\|\mathbf{x} - \mathbf{y}\|_2 \quad \forall \mathbf{x}, \mathbf{y} \in X. \qquad (8)$$

Then, after the stopping of Algorithm 1, the following inequality holds:

$$\min_{k\in I} f(\mathbf{x}^k) - f(\mathbf{x}^*) \leq M_f\varepsilon.$$

Proof. Note that

$$\min_{k\in I} f(\mathbf{x}^k) - f(\mathbf{x}^*) \leq \min_{k\in I} v_f(\mathbf{x}^k, \mathbf{x}^*) \cdot \|\nabla f(\mathbf{x}^k)\|_2 \leq M_f\varepsilon.$$

Now, we estimate the rate of convergence of Algorithm 1 for a differentiable objective functional f with a Lipschitz-continuous gradient.

$$\|\nabla f(x) - \nabla f(y)\|_* \leqslant L\|x - y\| \quad \forall x, y \in X. \qquad (9)$$

Assume that similarly to Devolder et al. (2014) we have an inexact (δ, L)-gradient $\nabla_\delta f$ for f:

$$f(\mathbf{x}) \leqslant f(\mathbf{x}_*) + \langle\nabla_\delta f(\mathbf{x}_*), \mathbf{x} - \mathbf{x}_*\rangle + \frac{1}{2}L\|\mathbf{x} - \mathbf{x}_*\|^2 + \delta$$

for exact solution $\mathbf{x}_*$ we can get that

$$\begin{aligned}&\min_{k\in I} f(\mathbf{x}^k) - f(\mathbf{x}_*)\\ &\leqslant \min_{k\in I}\left\{\|\nabla_\delta f(\mathbf{x}_*)\|_2\|\mathbf{x}^k - \mathbf{x}_*\| + \frac{1}{2}L\|\mathbf{x}^k - \mathbf{x}_*\|^2 + \delta\right\}.\end{aligned}$$

From Theorem 2 in view of Lemma 1 we obtain the following corollary

Corollary 4. Let f be differentiable and 9 hold. Assume that we have an inexact (δ, L)-gradient $\nabla_\delta f$ of function f at each point x. Then, after the stopping of Algorithm 1, the next inequality holds:

$$\min_{k\in I} f(\mathbf{x}^k) - f(\mathbf{x}_*) \leqslant \varepsilon\|\nabla f(\mathbf{x}_*)\|_2 + \frac{1}{2}L\varepsilon^2 + \delta.$$

Let us consider the rate of convergence of Algorithm 1 for a differentiable objective Hölder-continuous functional f, i.e. for some $\nu \in [0; 1)$

$$|f(x) - f(y)| \leqslant M_{f,\,\nu}\|x - y\|^\nu \quad \forall x, y \in Q. \qquad (10)$$

For example, $\nu = 1/2$ for $f(x) = \sqrt{x}$. The following corollary holds (see Stonyakin et al. (2020), Remark 2)

Corollary 5. Let f be a Hölder-continuous functional and (10) hold. Then, after the stopping of Algorithm 1 for $\varepsilon < 1$ we have

$$f(\widehat{x}) - f^* \leqslant \widehat{M}\varepsilon$$

for some $\hat{M} > 0$.

So, for problems with a convex Hölder-continuous differentiable objective and convex Lipcshitz-continuous functional constraints we can achieve an ε-solution after $O\left(\frac{1}{\varepsilon^2}\right)$ iterations of Algorithm 1. This estimate is optimal due to its optimality on a significantly narrower class of problems with Lipschitz-continuous objective functionals Nemirovsky and Yudin (1983).

3.2 Second algorithm

Let us consider the following method for fixed accuracy $\varepsilon > 0$, initial approach x^0, Θ_0: $\|x^0 - x_*\|_2^2 \leqslant 2\Theta_0^2$ and Lipschitz-continuous functional constraint g:

$$|g(x) - g(y)| \leqslant M_g||x - y|| \; \forall x, y \in \mathbb{R}_+^n.$$

Theorem 6. Let $\varepsilon > 0$ be a fixed number and the stopping criterion of Algorithm 2 be satisfied. Then the following inequality is true:

$$f(\hat{\mathbf{x}}^N) - f(\mathbf{x}_*) \leqslant \varepsilon \text{ and}$$

$$g(\hat{\mathbf{x}}^N) \leqslant \frac{\varepsilon}{\sum_{k\in I} h_k}\sum_{k\in I} h_k||g(\mathbf{x}^k)||_2 \leqslant \varepsilon M_g,$$

where $\hat{\mathbf{x}}^N = \frac{1}{\sum_{k\in I} h_k}\sum\limits_{k\in I} h_k\mathbf{x}^k$.

Proof. We give only a sketch of the proof because the proof of this theorem mostly follows the proof of theorem 2.
1) If $k \in I$ (for productive steps),

$$\begin{aligned}&h_k(f(\mathbf{x}^k) - f(\mathbf{x}^*))\\ &\leq \frac{\varepsilon^2}{2}\cdot\frac{1}{||\nabla f(\mathbf{x}^k)||_2^2} + \frac{1}{2}\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 - \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_2^2.\end{aligned}$$

2) If $k \notin I$ inequality (7) holds.

Algorithm 2 Another variant of adaptive mirror descent for many constraints.

```
Require: $\varepsilon > 0, \Theta_0 : \ d(\mathbf{x}^*) = \frac{1}{2}\|\mathbf{x}^0 - \mathbf{x}^*\|_2 \leqslant \Theta_0^2$, $\mathbf{x}^0 = 0$
    - initial point.
$I =: \emptyset$
$N \leftarrow 0$
repeat
    if $g_j(\mathbf{x}^N) \leqslant \varepsilon\|\nabla g_j(\mathbf{x}^N)\|_2, \ \forall j \in \overline{1,m}$ then
        $\mathbf{x}^{N+1} = \left[\mathbf{x}^N - \frac{\varepsilon \nabla f(\mathbf{x}^N)}{\|\nabla f(\mathbf{x}^N)\|_2^2}\right]_+$
        // $h_N = \frac{\varepsilon}{\|\nabla f(x^N)\|_2^2}$
        $N \to I$
    else
        $(g_{j_N}(\mathbf{x}^N) > \varepsilon\|\nabla g_{j_N}(\mathbf{x}^N)\|_2), \ j_N \in \overline{1,m})$
        $\mathbf{x}^{N+1} = \left[\mathbf{x}^N - \frac{\varepsilon \nabla g_{j_N}(\mathbf{x}^N)}{\|\nabla g_{j_N}(\mathbf{x}^N)\|_2}\right]_+$
        // $h_N = \frac{\varepsilon}{\|\nabla g_{j_N}(\mathbf{x}^N)\|_2}$
    end if
    $N \leftarrow N + 1$
until
        $$\frac{2\Theta_0^2}{\varepsilon^2} \leqslant \sum_{k \in I} \frac{1}{\|\nabla f(x^k)\|_*^2} + |J|, \qquad (11)$$
        where $|J|$ — the number of unproductive steps
    (we denote by $|I|$ the number of productive steps, i.e.
    $|I| + |J| = N$).
Ensure: $\hat{\mathbf{x}}^N = \frac{1}{\sum_{k \in I} h_k} \sum_{k \in I} h_k \mathbf{x}^k$
```

3) Summing the inequalities and after fulfilling the criterion for stopping the algorithm (11):

$$\sum_{k \in I} h_k(f(\mathbf{x}^k) - f(\mathbf{x}^*)) \leqslant \varepsilon \sum_{k \in I} h_k,$$

where for $\hat{\mathbf{x}}^N := \sum\limits_{k \in I} \frac{h_k \mathbf{x}^k}{\sum_{k \in I} h_k}$ holds (12). Wherein $\forall k \in I \ g(\mathbf{x}^k) \leqslant \varepsilon||\nabla g(\mathbf{x}^k)||_2 \leqslant \varepsilon M_g$ and holds (12).

Let us estimate the number of iterations necessary to fulfill the stopping criterion (11) in the case of a Lipschitz-continuous objective functional

$$|f(x) - f(y)| \leqslant M_f||x - y||_2.$$

It is clear that $\forall k \in I \ ||\nabla f(x^k)||_2 \leqslant M_f$ and therefore

$$|J| + \sum_{k \in I} \frac{1}{||\nabla f(x^k)||_*^2} \geqslant |J| + \frac{|I|}{M_f^2} \geqslant (|I| + |J|)\frac{1}{\max\{1, M_f^2\}}.$$

This means that for

$$\geqslant \frac{2\Theta_0^2 \max\{1, M_f^2\}}{\varepsilon^2} \qquad (12)$$

the stopping criterion (11) is obviously fulfilled, that is, the desired accuracy is achieved in $O\left(\frac{1}{\varepsilon^2}\right)$ iterations.

3.3 Modification for the logarithm utility functions

Note that the most common utility functions for networks are logarithms, i.e. $u_k(x_k) = \log x_k$. However, the logarithm is not a Lipschitz function on $\mathbb{R}_+^n$, since its gradient is unlimited near zero. However, consider the following modification of Algorithm 1. We shift the boundary of the feasible set from zero, i.e. let $x_k \geq \varepsilon n$, $k = 1, \dots, n$. Then, by the definition of the gradient of the logarithm, the utility function will be Lipschitz-continuous with the constant $M_U = \frac{1}{\varepsilon}$, i.e.

$$||\nabla U(\mathbf{x})||_2 \leq \sum_{k=1}^{n} |u_k'(x_k)| \leq n \cdot \frac{1}{\varepsilon n} = \frac{1}{\varepsilon}.$$

Firstly, to solve this problem, we apply Algorithm 1 for $N = \left\lceil 2\frac{\Theta_0^2}{\varepsilon^4} \right\rceil$ with $h_k = \frac{\varepsilon^2}{\|\nabla f(\mathbf{x}^k)\|_2}$ for $k \in I$ and $h_k = \frac{\varepsilon^2}{\|\nabla g_{j_k}(\mathbf{x}^k)\|_2}$ for $k \notin I$. Then, we obtain the following estimation for the convergence rate

Corollary 7. After the $N = \left\lceil 2\frac{\Theta_0^2}{\varepsilon^4} \right\rceil$ steps of Algorithm 1, the following inequality holds:

$$\min_{k \in I} f(\mathbf{x}^k) - f(\mathbf{x}^*) \leq \varepsilon.$$

Moreover, let us estimate the convergence rate of Algorithm 2 applied to this problem. Using estimation (12) with max at $M_U = \frac{1}{\varepsilon}$ we obtain the following corollary.

Corollary 8. After the $N = \left\lceil 2\frac{\Theta_0^2}{\varepsilon^4} \right\rceil$ steps of Algorithm 2, the following inequality holds:

$$f(\hat{\mathbf{x}}^N) - f(\mathbf{x}_*) \leqslant \varepsilon$$

where $\hat{\mathbf{x}}^N = \frac{1}{\sum_{k \in I} h_k} \sum\limits_{k \in I} h_k \mathbf{x}^k$.

Note that the convergence rates of Algorithm 1 and Algorithm 2 are of the same order, but due to the adaptability of the stopping criterion, Algorithm 2 works better in practice. Moreover, Algorithm 2 does not require modification of steps.

4. ELLIPSOID METHOD

Consider the transition to the dual problem for (1). Let $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m) \in \mathbb{R}_+^m$ be a vector of dual multipliers , which can be interpreted as a compound price vector. Define dual objective function

$$\varphi(\boldsymbol{\lambda}) = \max_{\mathbf{x} \in \mathbb{R}_+^n} \left\{ \sum_{k=1}^{n} u_k(x_k) + \langle \boldsymbol{\lambda}, \mathbf{b} - \sum_{k=1}^{n} \mathbf{C}_k x_k \rangle \right\} =$$

$$= \langle \boldsymbol{\lambda}, \mathbf{b} \rangle + \sum_{k=1}^{n} (u_k(x_k(\boldsymbol{\lambda})) - \langle \boldsymbol{\lambda}, \ \mathbf{C}_k x_k(\boldsymbol{\lambda}) \rangle),$$

and users choose the optimal data rates x_k by solving the following optimization problem

$$x_k(\boldsymbol{\lambda}) = \operatorname*{argmax}_{x_k \in \mathbb{R}_+} \{u_k(x_k) - x_k\langle \boldsymbol{\lambda}, \ \mathbf{C}_k \rangle\}. \qquad (13)$$

Then to find the optimal prices $\boldsymbol{\lambda}^*$ we need to solve the problem

$$\min_{\boldsymbol{\lambda} \in \mathbb{R}_+^m} \varphi(\boldsymbol{\lambda}). \qquad (14)$$

Suppose that for the primal problem the Slater condition is satisfied, then due to the strong duality both the primal and the dual problems will have a solution. Using Slater's condition, one can compactify the solution of the dual problem. We assume that the following estimate is correct for solving the dual problem:

$$||\boldsymbol{\lambda}^*||_2 \leq R.$$

In this case, the value R does not affect the operation of the algorithms under consideration, but R is only present in their convergence rate estimations.

To solve the dual problem, we consider the ellipsoid method. As the starting point of the method, we take the zero vector, i.e. $\boldsymbol{\lambda}^0 = 0$. The problem will be solved on the set Λ_{2R}, where

$$\Lambda_{2R} = \{\boldsymbol{\lambda} \in \mathbb{R}^m_+ : \|\boldsymbol{\lambda}\|_2 \le 2R\}.$$

Algorithm 3 Ellipsoid method

```
Require: $u_k(x_k), k = 1, ..., n$ — concave functions.
$B_0 := 2R \cdot I_n$
for $t = 0, ..., N-1$ do
    Compute $\nabla\varphi(\boldsymbol{\lambda}^t)$
    $\mathbf{q}_t := B_t^T \nabla\varphi(\boldsymbol{\lambda}^t)$
    $\mathbf{p}_t := \frac{B_t^T \mathbf{q}_t}{\sqrt{\mathbf{q}_t^T B_t B_t^T \mathbf{q}_t}}$
    $B_{t+1} := \frac{m}{\sqrt{m^2-1}} B_t + \left(\frac{m}{m+1} - \frac{m}{\sqrt{m^2-1}}\right) B_t \mathbf{p}_t \mathbf{p}_t^T$
    $\boldsymbol{\lambda}^{t+1} := \boldsymbol{\lambda}^t - \frac{1}{m+1} B_t \mathbf{p}_t$
end for
return $\boldsymbol{\lambda}^N$
```

To restore the solution of the primal problem by the solution of the dual problem, it is necessary to determine the accuracy certificate ξ for the ellipsoid method. The accuracy certificate is a sequence of weights $\xi = \{\xi_t\}_{t=0}^{N-1}$ such that

$$\xi_t \ge 0, \sum_{t=0}^{N-1} \xi_t = 1.$$

A detailed description of the construction of such a certificate can be found in Nemirovski et al. (2010).

Now we formulate a theorem of convergence rate estimation Ivanova et al. (2019).

Theorem 9. Let Algorithm 3 start with an initial ball $B_0 = \{\boldsymbol{\lambda} \in \mathbb{R}^m : \|\boldsymbol{\lambda}\|_2 \le 2R\}$. Then after

$$N = 2m(m+1)\left\lceil \log\left(\frac{32 \cdot 4MR}{\varepsilon}\right)\right\rceil$$

the following inequalities will hold

$$U(\mathbf{x}^*) - U(\hat{\mathbf{x}}^N) \le \varepsilon, \; \|[C\hat{\mathbf{x}}^N - \mathbf{b}]_+\|_2 \le \varepsilon,$$

where $\hat{\mathbf{x}}^N = \sum_{t \in I_N} \xi_t \mathbf{x}(\boldsymbol{\lambda}^t), \; I_N = \{t \le N-1 : \boldsymbol{\lambda}^t \in \text{int}\,\Lambda_{2R}\}$.

5. EXPERIMENTS

To test the performance of Algorithm 2 we compared it with the ellipsoid method (Algorithm 3). The behavior of the methods was tested in problems (1) of different configurations of networks and with different accuracy ε.

The routing matrix C was generated as follows: $C_i^j = 1$ with probability $p = 0.5$ or $C_i^j = 0$ with probability $1 - p = 0.5$. The elements of the vector $\mathbf{b}$ are uniform random variables: $b_i \in [0.1, 0.4]$. The utility functions are logarithmic. The initial values for Algorithm 2 and the ellipsoid method (EM) are $\mathbf{x}^0 = 0$ and $\boldsymbol{\lambda}^0 = 10^{-20}$, respectively. The radius $2R$ of the initial ball in the ellipsoid method and the radius $R = \sqrt{2}\Theta$ of the ball containing $\mathbf{x}^*$ in Algorithm 2 were chosen experimentally in such a way that the intermediate solutions obtained by the methods remained inside the given set at each iteration. The required solution accuracy ε was chosen so that the boundary shift $n\varepsilon$ of the feasible set from zero was small enough, that is no more than $\sim 10^{-1}$.

Table 1. Convergence results of Algorithm 2 (A2) and the ellipsoid method (EM), $\varepsilon = 6e-4$

	n	50		100		200	
	m	100	150	100	150	100	150
A2	Iter	142243	142516	171292	174270	193621	198585
	Time, s	16.77	21.91	33.56	37.63	46.8	49.22
EM	Iter	512749	758327	531448	760537	532992	761008
	Time, s	601.74	885.54	1022.73	1293.15	1418.67	1481.02

Table 2. Convergence results of Algorithm 2 (A2) and the ellipsoid method (EM), $\varepsilon = 3e-4$

	n	50		100		200	
	m	100	150	100	150	100	150
A2	Iter	8724510	9105234	9006192	9574296	9157003	9611472
	Time, s	921.38	1224.70	1276.73	1411.67	1424.12	1670.74
EM	Iter	603578	801775	628267	833323	633571	850051
	Time, s	1084.22	1317.61	1321.48	1705.46	1492.88	1921.06

Table 3. Convergence results of Algorithm 2 (A2) and the ellipsoid method (EM), $\varepsilon = 2e-4$

	n	50		100		200	
	m	100	150	100	150	100	150
A2	Iter	25225735	29752323	26055762	33846145	28532359	37244837
	Time, s	1367.54	1569.25	1550.62	1796.34	1723.07	1985.52
EM	Iter	599423	960529	618783	971525	667294	1021528
	Time, s	1223.86	1515.32	1677.25	1900.03	1891.18	2293.34

The results of the experiments are presented in Tables 1-3. As one can see from Tables 1-2, for $\varepsilon = 6e-4$ and $\varepsilon = 3e-4$, the proposed algorithm shows better time than the ellipsoid method. Even for high solution accuracy, $\varepsilon = 2e-4$, Algorithm 2 showed a large number of iterations and almost the same time as EM, as shown in Table 3. So, in a case where very high solution accuracy is not required, it is reasonable to apply the proposed algorithm.

The conducted experiments confirm the following theoretical fact about Algorithm 2: the convergence rate of Algorithm 2 depends only on the smoothness level of the target function and of the constraints and does not depend on the number of constraints m (see Section 3). Unlike in the case of the ellipsoid method, where there is a quadratic growth over m in the theoretical number of iterations (Theorem 9). So, if we compare the number of iterations (*Iter*) for the same n and for $m = 100$ and $m = 150$ in Tables 1-3, one can notice that the number of iterations for EM increases almost $1.5 - 2$ times and it is not the same for Algorithm 2. The theoretical results tell us that for Algorithm 2, as m increases and n is the same, the number of iterations should not change. But due to the adaptability of the stopping criterion, in practice it changes slightly, since in both cases this number is less than the theoretical convergence rate estimate.

6. CONCLUSION

In conclusion, we note that despite the theoretical attractiveness of Algorithm 1 (for example, as can be seen from Section 3.1, one can obtain estimates for cases with an inexact oracle and objective functions with different smoothness levels), our experiments showed that Algorithm 2 is significantly faster (both in time and in number of iterations) in practice due to adaptability of the stopping criterion.

Moreover, we considered Algorithm 3 from Stonyakin et al. (2018) (see also Algorithm 1 from Bayandina et al. (2018a)). Note, in comparison with Algorithm 2 this algorithm guarantees a better estimate for the residual by constraint $g(\mathbf{x}) \leq \varepsilon$ with similar estimates for the objective function and similar complexity $O\left(\varepsilon^{-2}\right)$. However, in practice, it works much worse than Algorithm 2 and Algorithm 1.

Another important observation concerning the comparison of Algorithms 2 and 1 with methods from Bayandina et al. (2018a); Stonyakin et al. (2018, 2019) in practice is that for medium and large networks. The practical result in a reasonable time can be obtained only from Algorithm 2.

In conclusion we note that those approaches considered by us allow us to determine the prices of connections. Here you can use the properties of primal-duality of considered Mirror Descent methods Bayandina et al. (2018b,a).

ACKNOWLEDGEMENTS

The authors are very grateful to P. Dvurechensky and Yu. Nesterov for fruitful discussions.

REFERENCES

Arrow, K.J. and Hurwicz, L. (1958). *Decentralization and computation in resource allocation.* Stanford University, Department of Economics.

Bayandina, A., Dvurechensky, P., Gasnikov, A., Stonyakin, F., and Titov, A. (2018a). Mirror descent and convex optimization problems with non-smooth inequality constraints. In *Large-Scale and Distributed Optimization: Lect. Notes Math*, volume 2227, 181–213. Springer, Cham.

Bayandina, A.S., Gasnikov, A.V., Gasnikova, E.V., and Matsievsky, S.V. (2018b). Primal-dual mirror descent for the stochastic programming problems with functional constraints. *Comput. Math. Math. Phys.*, 58(11), 1728–1736.

Campbell, D.E., Donald Edward, C., et al. (1987). *Resource allocation mechanisms.* Cambridge University Press.

Dehghan, M., Massoulie, L., Towsley, D., Menasche, D., and Tay, Y.C. (2016). A utility optimization approach to network cache design. In *IEEE INFOCOM 2016 — The 35th Annual IEEE International Conference on Computer Communications*, 1–9.

Devolder, O., Glineur, F., and Nesterov, Y. (2014). First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146(1-2), 37–75.

Friedman, E.J. and Oren, S.S. (1995). The complexity of resource allocation and price mechanisms under bounded rationality. *Economic Theory*, 6(2), 225–250.

Ivanova, A., Dvurechensky, P., Gasnikov, A., and Kamzolov, D. (2020). Composite optimization for the resource allocation problem. *Optimization Methods and Software*, doi: 10.1080/10556788.2020.1712599.

Ivanova, A., Pasechnyuk, D., Dvurechensky, P., Gasnikov, A., and Vorontsova, E. (2019). Numerical methods for the resource allocation problem in networks. *arXiv preprint arXiv:1909.13321.*

Kakhbod, A. (2013). *Resource allocation in decentralized systems with strategic agents: an implementation theory approach.* Springer Science & Business Media.

Kelly, F.P., Maulloo, A.K., and Tan, D.K.H. (1998). Rate control for communication networks: shadow prices, proportional fairness and stability. *J Oper Res Soc*, 49(3), 237–252.

Nemirovski, A., Onn, S., and Rothblum, U.G. (2010). Accuracy certificates for computational problems with convex structure. *Mathematics of Operations Research*, 35(1), 52–78.

Nemirovsky, A.S. and Yudin, D.B. (1983). Problem complexity and method efficiency in optimization.

Nesterov, Y. and Shikhman, V. (2018). Dual subgradient method with averaging for optimal resource allocation. *European Journal of Operational Research*, 270(3), 907–916.

Nesterov, Y. (2018). *Lectures on convex optimization*, volume 137. Springer.

Palomar, D.P. and Chiang, M. (2006). A tutorial on decomposition methods for network utility maximization. *IEEE Journal on Selected Areas in Communication*, 24(8), 1439–1451.

Rokhlin, D.B. (2019). Resource allocation in networks with a large number of users: stochastic gradient descent method. *Theory of Probability and Applications (in press)*, (in Russian).

Shakkottai, S. and Srikant, R. (2008). Network optimization and control. *Foundations and Trends in Networking*, (2(3)), 271–379.

Stonyakin, F.S., Alkousa, M.S., Stepanov, A.N., and Barinov, M.A. (2018). Adaptive mirror descent algorithms in convex programming problems with Lipschitz constraints. *Trudy Instituta Matematiki i Mekhaniki URO RAN*, 24(2), 266–279.

Stonyakin, F.S., Alkousa, M.S., Stepanov, A.N., and Titov, A.A. (2019). Adaptive mirror descent algorithms for convex and strongly convex optimization problems with functional constraints. *J. Appl. Ind. Math.*, 13, 557–574.

Stonyakin, F.S., Stepanov, A.N., Gasnikov, A.V., and Titov, A.A. (2020). Mirror descent for constrained optimization problems with large subgradient values of functional constraints. *Computer Research and Modeling*, 12(2), 301–317.