COMPUTER RESEARCH AND MODELING
2023 VOL. 15 NO. 2 P. 469–480
DOI: 10.20537/2076-7633-2023-15-2-469-480

Ки&М

MATHEMATICAL MODELING AND NUMERICAL SIMULATION

UDC: 519.8

Nonsmooth Distributed Min-Max Optimization Using the Smoothing Technique

J. Chen[a], A. V. Lobanov[b], A. V. Rogozin[c]

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

E-mail: [a] chen.ts@phystech.edu, [b] lobbsasha@mail.ru, [c] aleksandr.rogozin@phystech.edu

Received 19.02.2023.
Accepted for publication 23.02.2023.

Distributed saddle point problems (SPPs) have numerous applications in optimization, matrix games and machine learning. For example, the training of generated adversarial networks is represented as a min-max optimization problem, and training regularized linear models can be reformulated as an SPP as well. This paper studies distributed nonsmooth SPPs with Lipschitz-continuous objective functions. The objective function is represented as a sum of several components that are distributed between groups of computational nodes. The nodes, or agents, exchange information through some communication network that may be centralized or decentralized. A centralized network has a universal information aggregator (a server, or master node) that directly communicates to each of the agents and therefore can coordinate the optimization process. In a decentralized network, all the nodes are equal, the server node is not present, and each agent only communicates to its immediate neighbors.

We assume that each of the nodes locally holds its objective and can compute its value at given points, i. e. has access to zero-order oracle. Zero-order information is used when the gradient of the function is costly, not possible to compute or when the function is not differentiable. For example, in reinforcement learning one needs to generate a trajectory to evaluate the current policy. This policy evaluation process can be interpreted as the computation of the function value. We propose an approach that uses a smoothing technique, i. e., applies a first-order method to the smoothed version of the initial function. It can be shown that the stochastic gradient of the smoothed function can be viewed as a random two-point gradient approximation of the initial function. Smoothing approaches have been studied for distributed zero-order minimization, and our paper generalizes the smoothing technique on SPPs.

Keywords: convex optimization, distributed optimization

Citation: *Computer Research and Modeling*, 2023, vol. 15, no. 2, pp. 469–480.

This work was supported by the Ministry of Science and Higher Education of the Russian Federation (Goszadaniye) 075-00337-20-03, project No. 0714-2020-0005.

**КОМПЬЮТЕРНЫЕ ИССЛЕДОВАНИЯ
И МОДЕЛИРОВАНИЕ 2023 Т. 15 № 2 С. 469–480**
DOI: 10.20537/2076-7633-2023-15-2-469-480

МАТЕМАТИЧЕСКИЕ ОСНОВЫ И ЧИСЛЕННЫЕ МЕТОДЫ МОДЕЛИРОВАНИЯ

УДК: 519.8

Решение негладких распределенных минимаксных задач с применением техники сглаживания

Ц. Чэнь[a], А. В. Лобанов[b], А. В. Рогозин[c]

Московский физико-технический институт (национальный исследовательский университет),
Россия, 141701, Московская обл., г. Долгопрудный, Институтский пер., 9

E-mail: [a] chen.ts@phystech.edu, [b] lobbsasha@mail.ru, [c] aleksandr.rogozin@phystech.edu

Получено 19.02.2023.
Принято к публикации 23.02.2023.

Распределенные седловые задачи имеют множество различных приложений в оптимизации, теории игр и машинном обучении. Например, обучение генеративных состязательных сетей может быть представлено как минимаксная задача, а также задача обучения линейных моделей с регуляризатором может быть переписана как задача поиска седловой точки. В данной статье исследуются распределенные негладкие седловые задачи с липшицевыми целевыми функциями (возможно, недифференцируемыми). Целевая функция представляется в виде суммы нескольких слагаемых, распределенных между группой вычислительных узлов. Каждый узел имеет доступ к локально хранимой функции. Узлы, или агенты, обмениваются информацией через некоторую коммуникационную сеть, которая может быть централизованной или децентрализованной. В централизованной сети есть универсальный агрегатор информации (сервер или центральный узел), который напрямую взаимодействует с каждым из агентов и, следовательно, может координировать процесс оптимизации. В децентрализованной сети все узлы равноправны, серверный узел отсутствует, и каждый агент может общаться только со своими непосредственными соседями.

Мы предполагаем, что каждый из узлов локально хранит свою целевую функцию и может вычислить ее значение в заданных точках, т. е. имеет доступ к оракулу нулевого порядка. Информация нулевого порядка используется, когда градиент функции является трудно вычислимым, а также когда его невозможно вычислить или когда функция не дифференцируема. Например, в задачах обучения с подкреплением необходимо сгенерировать траекторию для оценки текущей стратегии. Этот процесс генерирования траектории и оценки политики можно интерпретировать как вычисление значения функции. Мы предлагаем подход, использующий технику сглаживания, т. е. применяющий метод первого порядка к сглаженной версии исходной функции. Можно показать, что стохастический градиент сглаженной функции можно рассматривать как случайную двухточечную аппроксимацию градиента исходной функции. Подходы, основанные на сглаживании, были изучены для распределенной минимизации нулевого порядка, и наша статья обобщает метод сглаживания целевой функции на седловые задачи.

Ключевые слова: выпуклая оптимизация, распределенная оптимизация

Работа выполнена при поддержке Министерства науки и высшего образования Российской Федерации (госзадание) № 075-00337-20-03, номер проекта 0714-2020-0005.

Introduction

We consider sum-type saddle point problems (SPP) of the form

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x, y) := \frac{1}{m}\sum_{i=1}^{m} f_i(x, y). \tag{1}$$

Each f_i is stored at a separate computational node that communicates through some network, either centralized or decentralized. We assume that functions f_i are convex and possibly nonsmooth. Moreover, each node can compute the value of function $f_i(x, y)$ at a given point. In other words, each node has access to a zero-order oracle of $f_i(x, y)$.

Related work

Saddle-point problems (SPP) have numerous applications in optimization and machine learning. For example, the development of generative adversarial networks (GANs) [Goodfellow et al., 2020] and reinforcement learning [Jin, Sidford, 2020] have fueled the interest of machine learning community in solving SPP. Moreover, saddle-point reformulation arises in training linear models with regularizers, see, i. e., [Kovalev, Gasnikov, Richtárik, 2021] and references therein.

The gradient and hessian (i. e., higher order characteristics) of $f_i(x, y)$ are not accessible. The type of optimization methods that work only with function values are referred to as zero-order, or gradient-free methods [Conn, Scheinberg, Vicente, 2009]. A typical approach to optimization of functions with zero-order characteristics is gradient approximation using function values [Nesterov, Spokoiny, 2017]. A residual gradient scheme can also be represented as a smoothing technique [Shamir, 2017]. The nonsmooth function is replaced by its smoothed version that is differentiable. For function smoothing, different distributions may be used, i. e., uniform over the Euclidean ball [Gasnikov et al., 2022] or Gaussian [Scaman et al., 2018].

It is also worth mentioning that nonsmooth problems may be solved by subgradient methods [Nesterov, 2003]. Subgradient methods are not zero-order, since the subgradient represents first-order information about a function. Distributed variants of subgradient algorithms were proposed in [Nedić, Ozdaglar, 2009; Forero, Cano, Giannakis, 2010].

Our paper focuses on distributed optimization. The objective functions are distributed over nodes in the network that may have a centralized aggregator or be fully decentralized. For surveys of distributed optimization in machine learning and other problems, see [Nedic, 2020; Rogozin et al., 2022]. Distributed methods for min-max problems with corresponding lower bounds were studied in [Beznosikov, Samokhin, Gasnikov, 2020]. Paper [Kovalev et al., 2022] proposed an optimal method for decentralized SPP solution with variance reduction.

Our contribution

In this work, we apply a smoothing technique [Shamir, 2017; Gasnikov et al., 2022]. The non-smooth function is replaced with its smoothed variant that is differentiable. The smoothing is performed over a Euclidean ball. Similarly to [Shamir, 2017; Gasnikov et al., 2022], we show that the stochastic gradient of the smoothed function can be interpreted as a two-point gradient approximation using function values [Nesterov, Spokoiny, 2017; Beznosikov, Sadiev, Gasnikov, 2020]. After that, the resulting method is based on the stochastic extra-gradient and the analysis of [Beznosikov, Samokhin, Gasnikov, 2020]. As a result, we obtain centralized and decentralized algorithms for min-max nonsmooth problems that use a smoothing technique.

Paper organization

In the section “Definitions and conditions” we introduce notation, definitions and conditions. After that, in the section “Smoothing Technique”, we show how to apply a smoothing technique to our optimization task. In the section “Zero-order Centralized and Decentralized Extra Step Methods” we give our main results and complexity bounds. Finally, we provide a discussion and make concluding remarks in the section “Discussion and Future Work”.

Definitions and conditions

Notation

Let $\langle x, y\rangle := \sum\limits_{i=1}^{d} x_i y_i$ define the inner product of $x, y \in \mathbb{R}^d$. Also let $\|x\|_p := \left(\sum\limits_{i=1}^{d} |x_i|^p\right)^{1/p}$ denote l_p-norm ($p \geqslant 1$) in $\mathbb{R}^d$. Let $B_p^d(r) := \left\{x \in \mathbb{R}^d \colon \|x\|_p \leqslant r\right\}$ denote a l_p-ball in $\mathbb{R}^d$ with center at 0 and radius r (the dimension is known from the context). Analogously, denote by $S_p^d(r) :=$ $:= \left\{x \in \mathbb{R}^d \colon \|x\|_p = r\right\}$ a l_p-sphere in $\mathbb{R}^d$. Also, let d_x and d_y denote the dimensions of x and y, respectively. Let $d := d_x + d_y$, introduce $z = (x^\top, y^\top)^\top \in \mathbb{R}^d$, and denote $\varphi(z) := f(x, y)$. We write $\xi \sim \mathcal{U}(Q)$ for a random variable ξ uniformly distributed over the set Q. Let $\partial_x(\cdot)$, $\partial_y(\cdot)$ denote a subdifferential of f in variables x and y, respectively.

We measure the quality of the solution of (1) using a duality gap.

Definition 1. Define the duality gap at point $(\widehat{x}, \widehat{y}) \in \mathcal{X} \times \mathcal{Y}$ as

$$\operatorname{gap}(\widehat{x}, \widehat{y}) = \max_{x\in\mathcal{X}} f(x, \widehat{y}) - \min_{y\in\mathcal{Y}} f(\widehat{x}, y).$$

We begin by stating the main assumptions of the paper.

Condition 1. *For each $i = 1, \ldots, m$ the function $f_i(x, y)$ is convex in x and concave in y.*

Condition 2. *For each $i = 1, \ldots, m$ the function $f_i(x, y)$ is μ-strongly convex in x and μ-strongly concave in y.*

We impose the following assumption that is standard for nonsmooth optimization.

Condition 3. *For each $i = 1, \ldots, m$ the function $f_i(x, y)$ is M_x-Lipschitz in x and M_y-Lipschitz in y w.r.t. the l_p-norm. That is, for all $x, x_1, x_2 \in \mathcal{X}$ and $y, y_1, y_2 \in \mathcal{Y}$ we have*

$$|f_i(x_2, y) - f_i(x_1, y)| \leqslant M_x \left\|x_2 - x_1\right\|_p,$$
$$|f_i(x, y_2) - f_i(x, y_1)| \leqslant M_y \left\|y_2 - y_1\right\|_p.$$

In particular, for $p = 2$ we use the notation $M_{2,x}$ and $M_{2,y}$ for the corresponding Lipschitz constants in l_2-norm.

Condition 4. *For each $i = 1, \ldots, m$ the function $f_i(x, y)$ is $M_{2,x}$-Lipschitz in x and $M_{2,y}$-Lipschitz in y w.r.t. the Euclidean norm. That is, for all $x, x_1, x_2 \in \mathcal{X}$ and $y, y_1, y_2 \in \mathcal{Y}$ we have*

$$|f_i(x_2, y) - f_i(x_1, y)| \leqslant M_{2,x} \left\|x_2 - x_1\right\|_2,$$
$$|f_i(x, y_2) - f_i(x, y_1)| \leqslant M_{2,y} \left\|y_2 - y_1\right\|_2.$$

For brevity we also introduce

$$M = \max(M_x, M_y), \quad M_2 = \max(M_{2,x}, M_{2,y}). \tag{2}$$

Smoothing technique

In this section, we present the basic elements of the idea described in detail in [Gasnikov et al., 2022, Section 2]. The main idea is to switch to the solution of the smooth optimization problem. That is, to solve a nonsmooth optimization problem (1) with ε-accuracy, it is enough to solve a smooth optimization problem with $\frac{\varepsilon}{2}$-accuracy.

Below is a detailed presentation of the connection between the two problems (smooth and nonsmooth optimization problems).

Smoothing approximation

For each $i = 1, \ldots, m$ we introduce a smooth approximation of $f_i(x, y)$ over a Euclidean ball. Introduce $e = \left(e_x^\top, e_y^\top\right)^\top \in \mathbb{R}^d$ and let $e \sim \mathcal{U}\left(B_2^d(1)\right)$. We let r denote the smoothing parameter and introduce

$$\widetilde{\varphi}_i(z) := \mathbb{E}_e[\varphi_i(z + re)]. \tag{3}$$

The following lemma provides properties of the smoothed function (3).

Lemma 1. *Let Condition* 3 *hold. Then for* $\widetilde{\varphi}_i(z)$ *from* (3)*, the following holds*

1. $\varphi_i(z) - rM_{2,y} \leqslant \widetilde{\varphi}_i(z) \leqslant \varphi_i(z) + rM_{2,x}$.
2. $|\widetilde{\varphi}_i(z_1) - \widetilde{\varphi}_i(z_2)| \leqslant M\|z_1 - z_2\|_p$.
3. $\|\nabla\widetilde{\varphi}_i(z_1) - \nabla\widetilde{\varphi}_i(z_2)\|_q \leqslant \frac{\sqrt{2d}M}{r}\|z_1 - z_2\|_p$, *where q is defined by* $\frac{1}{p} + \frac{1}{q} = 1$.

Proof. First, note that, since $e \sim \mathcal{U}\left(B_2^d(1)\right)$, we have $\left\|e_x\right\|_2 \leqslant 1$, $\left\|e_y\right\|_2 \leqslant 1$. If we apply this technique separately for x and y variables, we need to choose the corresponding r and M. For the second inequality, since $\varphi(z)$ is concave with respect to y and is a $M_{2,x}$ Lipschitz function with respect to x, then

$$\begin{aligned}\mathbb{E}_e[f(x + re_x, y - re_y)] - f(x, y) = \mathbb{E}_e[f(x + re_x, y - re_y) - f(x + re_x, y) + f(x + re_x, y) - f(x, y)] \leqslant \\ \leqslant \mathbb{E}_e[\langle\partial_y f(x + re_x, y), -re_y\rangle + rM_{2,x}\|e_x\|] \leqslant rM_{2,x}.\end{aligned}$$

For the first inequality, since $\varphi(z)$ is convex with respect to x and is a $M_{2,y}$ Lipschitz function with respect to y,

$$\begin{aligned}\mathbb{E}_e[f(x + re_x, y - re_y)] - f(x, y) = \mathbb{E}_e[f(x + re_x, y - re_y) - f(x, y - re_y) + f(x, y - re_y) - f(x, y)] \geqslant \\ \geqslant \mathbb{E}_e[\langle\partial_x f(x, y - re_y), re_x\rangle - rM_{2,y}\|e_y\|] \geqslant -rM_{2,y}.\end{aligned}$$

Hence, we can get the first point:

$$f(x, y) - rM_{2,y} \leqslant \mathbb{E}_e[f(x + re_x, y - re_y)] \leqslant f(x, y) + rM_{2,x},$$

that is,

$$\varphi(z) - rM_{2,y} \leqslant \widetilde{\varphi}(z) \leqslant \varphi(z) + rM_{2,x}.$$

As for the second point, we can have ($M = \max\{M_x, M_y\}$)

$$\begin{aligned}&|\widetilde{\varphi}(z_1) - \widetilde{\varphi}(z_2)| = |\mathbb{E}_e[f(x_1 + re_x, y_1 - re_y) - f(x_2 + re_x, y_2 - re_y)]| = \\ &= |\mathbb{E}_e[f(x_1 + re_x, y_1 - re_y) - f(x_1 + re_x, y_2 - re_y) + f(x_1 + re_x, y_2 - re_y) - f(x_2 + re_x, y_2 - re_y)]| \leqslant \\ &\leqslant \mathbb{E}_e[|f(x_1 + re_x, y_1 - re_y) - f(x_1 + re_x, y_2 - re_y)|] + \mathbb{E}_e[|f(x_1 + re_x, y_2 - re_y) - f(x_2 + re_x, y_2 - re_y)|] \leqslant \\ &\leqslant \mathbb{E}_e[M_y\|y_1 - y_2\|_p] + \mathbb{E}_e[M_x\|x_1 - x_2\|_p] \leqslant \max\{M_x, M_y\}\mathbb{E}_e[\|z_1 - z_2\|_p] = M\|z_1 - z_2\|_p.\end{aligned}$$

For the third point, analogously to [Gasnikov et al., 2022] we obtain

$$\left\|\nabla\widetilde{\varphi}(z_1) - \nabla\widetilde{\varphi}(z_2)\right\|_q^2 \leqslant \frac{2dM_2^2}{r^2}\left\|z_1 - z_2\right\|_p^2.$$

Black-box oracle and gradient approximation via l_2-randomization

The gradient of a function $\widetilde{\varphi}(z)$ can be approximated by function estimates at two points. This approximation model is known as a two-point gradient approximation. Then using the central finite difference, we introduce a gradient approximation using l_2 randomization:

$$g(z, \xi, e) = \mathbb{E}_{\widetilde{e}}\left[\frac{d}{2r}\mathbb{E}_{\xi}[(\varphi(z + r\widetilde{e}, \xi) - \varphi(z - r\widetilde{e}, \xi)]\begin{pmatrix}\widetilde{e}_x\\ -\widetilde{e}_y\end{pmatrix}\right], \tag{4}$$

where $\widetilde{e} := (\widetilde{e}_x, -\widetilde{e}_y)^\top$ is a vector uniformly distributed on $S_2^d(1)$. Next, in Lemma 2, we write out the properties of the approximation of the gradient (4).

Lemma 2. *Let Condition* 3 *be satisfied, then for all $z \in \mathcal{Z}$ we have*

1. *$g(z, \xi, \widetilde{e})$ is an unbiased estimation of $\nabla\widetilde{\varphi}(z)$, that is, $\mathbb{E}_{\xi,\widetilde{e}}[g(z, \xi, \widetilde{e})] = \nabla\widetilde{\varphi}(z)$.*
2. *$g(z, \xi, e)$ has a bounded variance, that is, $\mathbb{E}_{\xi,e}\left[\|g(z, \xi, \widetilde{e})\|_q^2\right] \leqslant \frac{dM_2^2}{\sqrt{2}r^2}a_q^2$, where $a_q^2 = \min\{2q - - 1, 32\log d - 8\}d^{2/q-1}$, $\forall d \geqslant 3$.*

Proof. For the first point, let $w = r\widetilde{e}$, then from the definition of $\widetilde{\varphi}_r(z)$

$$\widetilde{\varphi}_r(z) = \frac{1}{V\left(B_2^d(r)\right)}\int\limits_{\|w\|_2 \leqslant r}\mathbb{E}_{\xi}[\varphi(z + w, \xi)]\,dw.$$

Since $\varphi(z)$ is continuous and $\widetilde{\varphi}_r(z)$ is differentiable and its gradient can be found:

$$\nabla\widetilde{\varphi}_r(z) = \frac{1}{V\left(B_2^d(r)\right)}\int\limits_{\|w\|_2 = r}\mathbb{E}_{\xi}[\varphi(z + w, \xi)]\frac{w}{\|w\|_2}\,d_rS(z),$$

where $d_rS(z)$ is an element from a spherical surface of radius r.

After normalization to the normalized area (the area of the whole sphere is taken 1), we have integration with respect to a uniformly distributed probability $d\sigma(\widetilde{e})$ on S_1.

$$\nabla\widetilde{\varphi}_r(z) = \frac{d}{r}\int\limits_{\|\widetilde{e}\|_2 = 1}\mathbb{E}_{\xi}[\varphi(z + r\widetilde{e}, \xi)]\,d\sigma(\widetilde{e}) = \mathbb{E}_{\widetilde{e}\sim RS_1^d(0)}\left[\frac{d\varphi(z + r\widetilde{e}, \xi)\cdot\widetilde{e}}{r}\right].$$

Since $\varphi(z + r\widetilde{e}, \xi)\cdot\widetilde{e}$ has the same distribution as $\varphi(z - r\widetilde{e}, \xi)\cdot\widetilde{e}$, we can get

$$\nabla\widetilde{\varphi}_r(z) = \mathbb{E}_{e\sim RS_1^d(0)}\left[\frac{d(\mathbb{E}_{\xi}[\varphi(z + r\widetilde{e}, \xi) - \varphi(z - r\widetilde{e}, \xi)])\widetilde{e}}{2r}\right] = \mathbb{E}_{\xi,\widetilde{e}\sim RS_1^d(0)}[g(z, \xi, \widetilde{e})].$$

For the second point,

$$\mathbb{E}[\|g(z, \xi, \widetilde{e})\|^2] = \mathbb{E}\left[\left\|\frac{d}{2r}(\varphi(z + r\widetilde{e}, \xi) - \varphi(z - r\widetilde{e}, \xi))\widetilde{e}\right\|^2\right] \leqslant \frac{d^2}{4r^2}\mathbb{E}\left[(\varphi(z + r\widetilde{e}, \xi) - \varphi(z - r\widetilde{e}, \xi))^2\|\widetilde{e}\|^2\right].$$

By the independence of $\xi, \widetilde{e}$, we have:

$$\mathbb{E}\left[\|g(z,\,\xi,\,e)\|^2\right] \leqslant \frac{d^2}{4r^2}\mathbb{E}_\xi\left[\mathbb{E}_e\left[(\varphi(z + r\widetilde{e},\,\xi) - \alpha - \varphi(z - r\widetilde{e},\,\xi) + \alpha)^2\|e\|^2\right]\right] \leqslant$$
$$\leqslant \frac{d^2}{2r^2}\mathbb{E}_\xi\left[\mathbb{E}_e\left[\left((\varphi(z + r\widetilde{e},\,\xi) - \alpha)^2 + (\varphi(z - r\widetilde{e},\,\xi) - \alpha)^2\right)\|\widetilde{e}\|^2\right]\right].$$

Using the symmetric distribution of $\widetilde{e}$, we have

$$\mathbb{E}\left[\|g(z,\,\xi,\,\widetilde{e})\|_q^2\right] \leqslant \frac{d^2}{r^2}\mathbb{E}_\xi\left[\mathbb{E}_{\widetilde{e}}\left[(\varphi(z + r\widetilde{e},\,\xi) - \alpha)^2\right]\|\widetilde{e}\|_q^2\right] \leqslant$$
$$\leqslant \frac{d^2}{r^2}\mathbb{E}_\xi\left[\sqrt{\mathbb{E}_{\widetilde{e}}[(\varphi(z + r\widetilde{e},\,\xi) - \alpha])^4}\,\sqrt{\mathbb{E}_{\widetilde{e}}\left[\|\widetilde{e}\|_q^4\right]}\right] \overset{①}{\leqslant} \frac{dM_2^2}{\sqrt{2}r^2}\sqrt{\mathbb{E}_{\widetilde{e}}\left[\|\widetilde{e}\|_q^4\right]} \leqslant \frac{dM_2^2}{\sqrt{2}r^2}a_q^2,$$

where ① holds according to the results in [Shamir, 2017; Lobanov et al., 2022]. Then we have:

$$\mathbb{E}_{\xi,\widetilde{e}}\left[\|g(z,\,\xi,\,\widetilde{e})\|_q^2\right] \leqslant \frac{dM_2^2}{\sqrt{2}r^2}a_q^2,$$

where $a_q^2 = \min\{2q - 1,\, 32\log d - 8\}d^{2/q-1}$, $\forall d \geqslant 3$. □

Zero-order Centralized and Decentralized Extra Step Methods

In this section, we present algorithms for solving the distributed optimization problem (1) using the smoothing technique.

Centralized case

The following Algorithm 1 is an adaptation of the algorithm from [Beznosikov, Samokhin, Gasnikov, 2020] for the nonsmooth problem (1) in the centralized case.

Algorithm 1. Zero-order centralized extra step method

```
Require: Stepsize $\gamma$, number of communication rounds $K$; number of local iterations $T$, batch size $b$
Ensure: Choose $\left(x^0,\, y^0\right) = z^0 \in Z$, batch size $b = \lfloor \frac{T}{2k} \rfloor$.
  for $t = 0$ to $k$ do
    agent $i$ computes $g_i^t = \frac{1}{b}\sum_{j=1}^{b} g_i\left(z^t,\, \xi^{t,j}, \mathbf{e}\right)$ and send $g_i^t$ to master server;
    sever compute $z^{t+1/2} = \mathrm{proj}\left(z^t - \frac{\gamma}{m}\sum_{i=1}^{m} g_i^t\right)$ and send $z^{t+1/2}$ to agent $i$.
    agent $i$ computes $g_i^{t+1/2} = \frac{1}{b}\sum_{j=1}^{b} g_i\left(z^{t+1/2},\, \xi^{t+1/2,j},\, \mathbf{e}\right)$ and send $g_i^{t+1/2}$ to master server;
    sever compute $z^{t+1} = \mathrm{proj}\left(z^t - \frac{\gamma}{m}\sum_{i=1}^{m} g_i^{t+1/2}\right)$ and send $z^{t+1}$ to agent $i$.
  end for
```

The convergence rate results of Algorithm 1 for the convex-concave and strongly convex-strongly concave cases are presented in Theorem 1.

Theorem 1. *Let Ω_z be the diameter of $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, ε be the accuracy of the solution to problem* (1) $\left(\textit{where } \mathbb{E}\left[gap\left(\overline{z}^K\right)\right] \leqslant \varepsilon\right)$*, and $r = \frac{\varepsilon}{2M_2}$ be the smoothing parameter, and let M and b be the numbers of the computation node, and the batch size, respectively. Denote by k the iteration number, then*

- *for the convex-concave function, Algorithm* 1 *has the following convergence rate*:

$$\mathbb{E}\left[gap\left(\overline{z}^K\right)\right] = O\left(\frac{\sqrt{d}M\Omega_z^2}{rK} + \frac{\sqrt{d}M_2 a_q \Omega_z}{r\sqrt{mT}}\right);$$

- *for the strongly convex-strongly concave function, Algorithm* 1 *has the following convergence rate*:

$$\mathbb{E}\left[\left\|z^{K+1} - z^\star\right\|\right] = \widetilde{O}\left(\left\|z_0 - z^\star\right\|^2 \exp\left(-\frac{\mu K r}{\sqrt{d}M}\right) + \frac{dM_2^2 a_q^2}{\mu^2 r^2 m T}\right).$$

Proof. To begin we introduce some results from [Beznosikov, Samokhin, Gasnikov, 2020], where γ is the step size, Ω_z is the space diameter, L is the constant of the Lipschitz continuous gradient, σ^2 is the upper bound of gradient variance and b, m are the batch size and the agent number, respectively.

1. For the smooth centralized convex and concave saddle point problem, we can have the convergence rate
$$\mathbb{E}\left(\overline{z}_{gap}^{K+1}\right) = \frac{\Omega_z^2}{2\gamma(k+1)} + \frac{9\gamma\sigma^2}{2bm}. \qquad (5)$$
2. For the smooth centralized strong convex and strong concave saddle point problem, we have below the inequalities
$$\mathbb{E}\left[\left\|z^{K+1} - z^\star\right\|\right] \leqslant (1-\mu\gamma)^K \mathbb{E}\left[\left\|z_0 - z^*\right\|^2\right] + \frac{6\sigma^2\gamma}{\mu bm}. \qquad (6)$$

Convex and concave

We consider the average gradient of $g_i(z, \xi, e)$ among m agents:

$$\overline{g}^{t+1/2} = \sum_{i=1}^m g_i\left(z^{t+1/2}, \xi, e\right), \quad \overline{g}^{t+1} = \sum_{i=1}^m g_i\left(z^{t+1}, \xi, e\right).$$

We use the duality gap of the saddle point problem to calculate the convergence rate, let $\overline{x}^{k+1} = \frac{1}{k+1}\sum_{t=0}^{k} x^{t+1/2}$, $\overline{y}^{k+1} = \frac{1}{k+1}\sum_{t=0}^{k} y^{t+1/2}$ and $\overline{z}^{t+1} = \left(\overline{x}^{k+1}, \overline{y}^{k+1}\right)$:

$$\begin{aligned}\mathbb{E}\left[\text{gap}\left(\overline{z}^{t+1}\right)\right] &= \max_{y'} \mathbb{E}_\xi\left[\varphi\left(\overline{x}^{t+1}, y', \xi\right)\right] - \min_{x'} \mathbb{E}_\xi\left[\varphi\left(x', \overline{y}^{t+1}, \xi\right)\right] = \\ &= \max_{y'} \varphi\left(\overline{x}^{t+1}, y'\right) - \min_{x'} \varphi\left(x', \overline{y}^{t+1}\right) \leqslant \max_{y'} \widetilde{\varphi}\left(\overline{x}^{t+1}, y'\right) - \min_{x'} \widetilde{\varphi}\left(x', \overline{y}^{t+1}\right) + r_x M_{2,x} + r_y M_{2,y} \leqslant \\ &\leqslant \max_{y'} \widetilde{\varphi}\left(\overline{x}^{t+1}, y'\right) - \min_{x'} \widetilde{\varphi}\left(x', \overline{y}^{t+1}\right) + rM_2 \leqslant \varepsilon.\end{aligned}$$

Thus, in order to solve problem (1) with ε-accuracy, we need to solve the smoothed problem with $\frac{\varepsilon}{2}$-accuracy. Then given that $rM_2 = \frac{\varepsilon}{2}$ $\left(r = \frac{\varepsilon}{2M_2}\right)$, we can estimate $\mathbb{E}\left[\text{gap}\left(\overline{z}^{t+1}\right)\right]$ for (5), substituting $L = \frac{\sqrt{d}M}{r}$ from Lemma 1 and $\sigma^2 = \frac{dM_2^2 a_q^2}{r^2}$ from Lemma 2 and $\gamma = \min\left\{\frac{1}{4L}; \frac{\Omega_z}{3\sigma}\sqrt{\frac{bm}{k+1}}\right\}$ from [Beznosikov, Samokhin, Gasnikov, 2020]:

$$\mathbb{E}\left[\text{gap}\left(\overline{z}^{K+1}\right)\right] = O\left(\frac{L\Omega_z^2}{K+1} + \frac{\Omega_z\sigma}{\sqrt{bm(K+1)}}\right) = O\left(\frac{\sqrt{d}M\Omega_z^2}{rK} + \frac{\sqrt{d}M_2 a_q \Omega_z}{r\sqrt{mT}}\right),$$

where b is the batch size from Algorithm 1.

Strong Convex and Strong Concave

In this case we apply the result from [Beznosikov, Samokhin, Gasnikov, 2020], which is for centralized smooth saddle point problem, then we replace the corresponding parameters to the smooth function in (6): $L = \frac{\sqrt{d}M}{r}$ from Lemma 1 and $\sigma^2 = \frac{dM_2^2a_q^2}{r^2}$ from Lemma 2. Then choosing $\gamma =$ $= \min\left\{\frac{1}{4L}, \frac{\ln\left(\max\left\{2; bm\mu^2\|z_0-z^\star\|^2\frac{k}{6\sigma}\right\}\right)}{\mu k}\right\}$, we obtain

$$\mathbb{E}\left[\left\|z^{K+1}-z^\star\right\|\right] = \widetilde{O}\left(\left\|z_0-z^\star\right\|^2\exp\left[-\frac{\mu K}{4L}\right]+\frac{\sigma^2}{\mu^2bmK}\right) =$$
$$= \widetilde{O}\left(\left\|z_0-z^\star\right\|^2\exp\left[-\frac{\mu Kr}{\sqrt{d}M}\right]+\frac{dM_2^2a_q^2}{r^2\mu^2bmK}\right) = \widetilde{O}\left(\left\|z_0-z^\star\right\|^2\exp\left[-\frac{\mu Kr}{\sqrt{d}M}\right]+\frac{dM_2^2a_q^2}{r^2\mu^2mT}\right),$$

where b is the batch size from Algorithm 1. □

Decentralized case

To solve the problem (1) in the decentralized case, we propose the following Algorithm 2.

Algorithm 2. Zero-order decentralized extra step method

```
Require: Stepsize $\gamma$; Communication round $K$; Number of local iteration $T$, number of FastMix
  steps $H$.
Ensure: Choose $\left(x^0, y^0\right) = z^0 \in Z$, $z_m^0 = z^0$, $k = \lfloor\frac{K}{H}\rfloor$, batch size $b = \lfloor\frac{T}{2k}\rfloor$.
  for $t = 1$ to $k$ do
    each agent $i$ compute $\widetilde{z}_i^{t+1/2} = z_i^t - \frac{\gamma}{b}\sum_{j=1}^{b} g_i\left(z_i^t, \xi_i^{t,j}, e\right)$;
    Communication: $\widetilde{z}_1^{t+1/2}, \ldots, \widetilde{z}_m^{t+1/2} = \mathrm{FastMix}\left(\widetilde{z}_1^{t+1/2}, \ldots, \widetilde{z}_m^{t+1/2}, H\right)$;
    Each agent $i$ compute: $z_i^{t+1/2} = \mathrm{proj}\left(\widetilde{z}_i^{t+1/2}\right)$;
    each agent $i$ compute: $\widetilde{z}_i^{t+1} = z_i^t - \frac{\gamma}{b}\sum_{j=1}^{b} g_i\left(z_i^{t+1/2}, \xi_i^{t+1/2,j}, e\right)$;
    Communication: $\widetilde{z}_1^{t+1}, \ldots, \widetilde{z}_m^{t+1} = \mathrm{FastMix}\left(\widetilde{z}_1^{t+1}, \ldots, \widetilde{z}_m^{t+1}, H\right)$;
    Each agent $i$ compute: $z_i^{t+1} = \mathrm{proj}\left(\widetilde{z}_i^{t+1}\right)$;
  end for
```

The following theorem presents the convergence rate results of Algorithm 2 for the decentralized case of problem (1).

Theorem 2. *Let Ω_z is the space diameter, d be the dimension, ε be the accuracy of the solution to problem* (1) $\left(\textit{where } \mathbb{E}\left[gap\left(\overline{z}^k\right)\right] \leqslant \varepsilon\right)$*, and let $r = \frac{\varepsilon}{2M_2}$ be the smoothing parameter, m and b be the numbers of the computation node, and the batch size, respectively. Denote by k the iteration number, then*

- *for the convex-concave function, we have the following convergence rate*:

$$\mathbb{E}\left[gap\left(\overline{z}^{k+1}\right)\right] = \widetilde{O}\left(\frac{\sqrt{d}M\Omega_z^2}{rK}+\frac{\Omega_zM_2a_q}{r}\sqrt{\frac{d}{bmT}}\right);$$

- *for the strongly convex-strongly concave function, we have the following convergence rate*:

$$\mathbb{E}\left[\left\|z^{k+1}-z^\star\right\|\right] = \widetilde{O}\left(\left\|z_0-z^\star\right\|^2\exp\left(-\frac{\mu Kr}{\sqrt{d}M}\right)+\frac{dM_2^2a_q^2}{r^2\mu^2mT}\right).$$

Proof. First of all, we present Algorithm 3 as an omitted procedure in Algorithm 2.

Algorithm 3. FastMix

```
Require: vectors $z_1, z_2, \ldots, z_m$ and communication rounds $H$;
Ensure: construct matrix $z$ with vectors $z_1, z_2, \ldots, z_m$, $z^{-1} = z$, $z^0 = z$, $\eta = \frac{1-\sqrt{1-\lambda_2^2(W)}}{1+\sqrt{1-\lambda_2^2(W)}}$;
  for $h = 0, 1, 2, \ldots, H$ do
    $z^{h+1} = (1+\eta)Wz^h - \eta z^{h-1}$
  end for
  return rows $z_1, z_2, \ldots, z_m$ of $z^H$
```

Next, we consider two cases that satisfy Conditions 1 and 2, respectively.

Convex and Concave

From [Beznosikov, Samokhin, Gasnikov, 2020], we have the convergence rate for the smooth case:

$$\mathbb{E}\left[\operatorname{gap}\left(\overline{z}^{t+1}\right)\right] = \widetilde{O}\left(\frac{L\Omega_z^2}{K} + \frac{\sigma\Omega_z}{\sqrt{mT}}\right). \tag{7}$$

Then by substituting $L = \frac{\sqrt{d}M}{r}$ from Lemma 1 and $\sigma^2 = \frac{dM_2^2a_q^2}{r^2}$ from Lemma 2 we obtain the convergence rate of Algorithm 2 for the initial problem (1):

$$\mathbb{E}\left[\operatorname{gap}\left(\overline{z}^{t+1}\right)\right] = \widetilde{O}\left(\frac{L\Omega_z^2}{K} + \frac{\sigma\Omega_z}{\sqrt{mT}}\right) = \widetilde{O}\left(\frac{\sqrt{d}M\Omega_z^2}{rK} + \frac{\Omega_z M_2 a_q}{r}\sqrt{\frac{d}{bmT}}\right). \tag{8}$$

Strong convex and Strong concave

From [Beznosikov, Samokhin, Gasnikov, 2020], we have the convergence rate for the smooth case:

$$\mathbb{E}\left[\|z^{t+1} - z^\star\|^2\right] = \widetilde{O}\left(\|z_0 - z^\star\|^2 \exp\left(-\frac{\mu K}{L}\right) + \frac{\sigma^2}{\mu^2 mT}\right). \tag{9}$$

Then by substituting $L = \frac{\sqrt{d}M}{r}$ from Lemma 1 and $\sigma^2 = \frac{dM_2^2a_q^2}{r^2}$ from Lemma 2 we obtain the convergence rate of Algorithm 2 for initial problem (1):

$$\mathbb{E}\left[\|z^{t+1} - z^\star\|^2\right] = \widetilde{O}\left(\|z_0 - z^\star\| 2^2 \exp\left(-\frac{\mu K}{L}\right) + \frac{\sigma^2}{\mu^2 mT}\right) = \widetilde{O}\left(\|z_0 - z^\star\|^2 \exp\left(-\frac{\mu K r}{\sqrt{d}M}\right) + \frac{dM_2^2a_q^2}{r^2\mu^2 mT}\right). \tag{10}$$

□

Discussion and future work

In this paper, we have considered distributed nonsmooth min-max optimization problems. The proposed method uses a smoothing technique that allows one to apply first-order methods to the smoothed function. The smoothing parameter is chosen proportionally to the required solution accuracy. The analysis of our method is similar to the analysis of first-order extra-step method for saddle point problems.

References

Bach F., Perchet V. Highly-smooth zero-th order online optimization // Conference on Learning Theory. — PMLR, 2016. — P. 257–283.

Beznosikov A., Novitskii V., Gasnikov A. One-point gradient-free methods for smooth and non-smooth saddle-point problems // Mathematical Optimization Theory and Operations Research: 20th International Conference, MOTOR 2021, Irkutsk, Russia, July 5–10, 2021. — Proceedings 20. — Springer, 2021. — P. 144–158.

Beznosikov A., Sadiev A., Gasnikov A. Gradient-free methods with inexact oracle for convex-concave stochastic saddle-point problem // Mathematical Optimization Theory and Operations Research: 19th International Conference, MOTOR 2020, Novosibirsk, Russia, July 6–10, 2020. — Revised Selected Papers 19. — Springer, 2020. — P. 105–119.

Beznosikov A., Samokhin V., Gasnikov A. Distributed saddle-point problems: Lower bounds, optimal and robust algorithms // arXiv preprint arXiv:2010.13112. — 2020.

Boyd S., Ghosh A., Prabhakar B., Shah D. Randomized gossip algorithms // IEEE transactions on information theory. — 2006. — Vol. 52, No. 6. — P. 2508–2530.

Bubeck S., Jiang Q., Lee Y. T., Li Y., Sidford A. Complexity of highly parallel non-smooth convex optimization // Advances in neural information processing systems. — 2019. — Vol. 32.

Conn A. R., Scheinberg K., Vicente L. N. Introduction to derivative-free optimization. — SIAM, 2009.

Diakonikolas J., Guzmán C. Lower bounds for parallel and randomized convex optimization // Conference on Learning Theory. — PMLR, 2019. — P. 1132–1157.

Duchi J. C., Jordan M. I., Wainwright M. J., Wibisono A. Optimal rates for zero-order convex optimization: The power of two function evaluations // IEEE Transactions on Information Theory. — 2015. — Vol. 61, No. 5. — P. 2788–2806.

Fazel M., Ge R., Kakade S., Mesbahi M. Global convergence of policy gradient methods for the linear quadratic regulator // International conference on machine learning. — PMLR, 2018. — P. 1467–1476.

Forero P. A., Cano A., Giannakis G. B. Consensus-based distributed linear support vector machines // Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks. — 2010. — P. 35–46.

Gasnikov A., Novitskii A., Novitskii V., Abdukhakimov F., Kamzolov D., Beznosikov A., Takáč M., Dvurechensky P., Gu B. The power of first-order smooth optimization for black-box non-smooth problems // arXiv preprint arXiv:2201.12289. — 2022.

Gasnikov A. V., Krymova E. A., Lagunovskaya A. A., Usmanova I. N., Fedorenko F. A. Stochastic online optimization. Single-point and multi-point non-linear multi-armed bandits. Convex and strongly-convex case // Automation and remote control. — 2017. — Vol. 78. — P. 224–234.

Goodfellow I., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A., Bengio Y. Generative adversarial networks // Communications of the ACM. — 2020. — Vol. 63, No. 11. — P. 139–144.

Gorbunov E., Dvurechensky P., Gasnikov A. An accelerated method for derivative-free smooth stochastic convex optimization // arXiv preprint arXiv:1802.09022. — 2018.

Jakovetic D., Xavier J., Moura J. M. Fast distributed gradient methods // IEEE Transactions on Automatic Control. — 2014. — Vol. 59, No. 5. — P. 1131–1146.

Jin Y., Sidford A. Efficiently solving MDPs with stochastic mirror descent // International Conference on Machine Learning. — PMLR, 2020. — P. 4890–4900.

Kovalev D., Beznosikov A., Sadiev A., Persiianov M., Richtárik P., Gasnikov A. Optimal algorithms for decentralized stochastic variational inequalities // arXiv preprint arXiv:2202.02771. — 2022.

Kovalev D., Gasnikov A., Richtárik P. Accelerated primal-dual gradient method for smooth and convex-concave saddle-point problems with bilinear coupling // arXiv preprint arXiv:2112.15199. — 2021.

Lobanov A., Alashqar B., Dvinskikh D., Gasnikov A. Gradient-Free Federated Learning Methods with l_1 and l_2-Randomization for Non-Smooth Convex Stochastic Optimization Problems // arXiv preprint arXiv:2211.10783. — 2022.

Nedic A. Distributed gradient methods for convex machine learning problems in networks: Distributed optimization // IEEE Signal Processing Magazine. — 2020. — Vol. 37, No. 3. — P. 92–101.

Nedić A., Ozdaglar A. Subgradient methods for saddle-point problems // Journal of optimization theory and applications. — 2009. — Vol. 142. — P. 205–228.

Nesterov Y. Introductory lectures on convex optimization: A basic course. — Springer Science & Business Media, 2003.

Nesterov Y., Spokoiny V. Random gradient-free minimization of convex functions // Foundations of Computational Mathematics. — 2017. — Vol. 17. — P. 527–566.

Rogozin A., Gasnikov A., Beznosikov A., Kovalev D. Decentralized optimization over time-varying graphs: a survey // arXiv preprint arXiv:2210.09719. — 2022.

Scaman K., Bach F., Bubeck S., Massoulié L., Lee Y. T. Optimal algorithms for non-smooth distributed optimization in networks // Advances in Neural Information Processing Systems. — 2018. — Vol. 31.

Shamir O. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback // The Journal of Machine Learning Research. — 2017. — Vol. 18, No. 1. — P. 1703–1713.