Распределенные и параллельные алгоритмы для задач анализа данных

Гасников А.В.
gasnikov.av@mipt.ru

МФТИ-Сириус; *21 апреля 2021*

Команда РФФИ

Защита докторской диссертации в декабре 2020 года

Гасников Александр Владимирович

руководитель

д.ф.-м.н. (2016), профессор кафедры Математических основ управления школы ПМИ МФТИ

Стонякин Фёдор Сергеевич

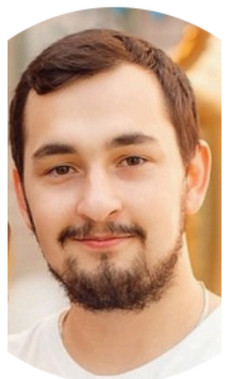
заместитель руководителя

д.ф.-м.н. (2020), доцент, научный сотрудник в МФТИ (основное место работы - Крымский федеральный университет им. В.И. Вернадского)

Защита кандидатской диссертации в июле 2021

Двинских Дарина Михайловна

аспирант университета Гумбольдта, Берлин; исследователь в школе ПМИ МФТИ



Горбунов Эдуард Александрович

аспирант школы ПМИ МФТИ

Защита кандидатской диссертации в октябре 2021

Команда РФФИ

Рогозин Александр Викторович

аспирант школы ПМИ МФТИ

Безносиков Александр Николаевич

студент магистратуры школы ПМИ МФТИ

Пасечнюк Дмитрий Аркадьевич

студент бакалавриата школы ПМИ МФТИ

http://dmivilensky.ru/opt/

Команда

Павел Двуреченский (WIAS, Berlin) д.ф.-м.н.

Дмитрий Камзолов (МФТИ) к.ф.-м.н.

Назарий Тупица (МФТИ) к.ф.-м.н.

Артем Агафонов (МФТИ) бакалавр

Structure of the Lecture

1. Smooth and nonsmooth convex optimization (Optimal methods)

2. Stochastic smooth and nonsmooth convex optimization (Optimal methods)

3. Parallelization of stochastic smooth convex optimization

4. Stochastic smooth and nonsmooth convex optimization under affine constraints

5. Primal and Dual oracle

6. Sum type target function

7. Applications the results of item 4 to decentralized distributed optimization with sum type target function (primal and dual oracles)

8. Further generalizations

Nonsmooth stochastic convex optimization

$$f(x) = \mathbb{E}[f(x, \xi)] \to \min_{x \in Q \subseteq \mathbb{R}^n}.$$

Oracle return unbiased stochastic gradient: $\nabla f(x, \xi)$

$$\mathbb{E}[\|\nabla f(x, \xi)\|_2^2] \leq M^2$$

$$R = \|x^0 - x^*\|_2$$

$$\forall x, y \in Q \qquad f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2}\|y - x\|_2^2.$$

$$\min\left\{O\left(\frac{M^2R^2}{\varepsilon^2}\right), O\left(\frac{M^2}{\mu\varepsilon}\right)\right\}$$ **— number of oracle calls**

ε **— precision in function values**

Smooth stochastic convex optimization

$$f(x) = \mathbb{E}[f(x, \xi)] \to \min_{x \in Q \subseteq \mathbb{R}^n}.$$

Oracle return unbiased stochastic gradient: $\nabla f(x, \xi)$

$$\mathbb{E}[\|\nabla f(x, \xi) - \nabla f(x)\|_2^2] \leq \sigma^2, \;\; R = \|x^0 - x^*\|_2$$

$$\forall x, y \in Q \qquad \|\nabla f(y) - \nabla f(x)\|_2 \leq L\|y - x\|_2.$$

$$\forall x, y \in Q \qquad f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2}\|y - x\|_2^2.$$

$$\min\left\{O\left(\sqrt{\frac{LR^2}{\varepsilon}}\right) + O\left(\frac{\sigma^2 R^2}{\varepsilon^2}\right), O\left(\sqrt{\frac{L}{\mu}}\ln\left(\frac{\mu R^2}{\varepsilon}\right)\right) + O\left(\frac{\sigma^2}{\mu\varepsilon}\right)\right\}$$

— number of oracle calls

Parallel smooth stochastic convex optimization

$$f(x) = \mathbb{E}[f(x,\xi)] \to \min_{x \in Q \subseteq \mathbb{R}^n}.$$

Oracle return unbiased stochastic gradient: $\nabla f(x,\xi)$

$$\min\left\{O\left(\sqrt{\frac{LR^2}{\varepsilon}}\right) + O\left(\frac{\sigma^2 R^2}{\varepsilon^2}\right), O\left(\sqrt{\frac{L}{\mu}}\ln\left(\frac{\mu R^2}{\varepsilon}\right)\right) + O\left(\frac{\sigma^2}{\mu\varepsilon}\right)\right\}$$

We can parallelize calculation on

$$O\left(\frac{\frac{\sigma^2 R^2}{\varepsilon^2}}{\sqrt{\frac{LR^2}{\varepsilon}}}\right) \text{ or } O\left(\frac{\frac{\sigma^2}{\mu\varepsilon}}{\sqrt{\frac{L}{\mu}}\ln\left(\frac{\mu R^2}{\varepsilon}\right)}\right)$$

processors.

Optimal algorithm for smooth convex optimization

Algorithm 1 Similar Triangles Method $\texttt{STM}(L, \mu, x^0)$, $Q = \mathbb{R}^n$

Input: $\tilde{x}^0 = z^0 = x^0$, number of iterations N, $\alpha_0 = A_0 = 0$, L, μ

1: **for** $k = 0, \ldots, N$ **do**

2: Set $\alpha_{k+1} = \frac{1+A_k\mu}{2L} + \sqrt{\frac{1+A_k\mu}{4L^2} + \frac{A_k(1+A_k\mu)}{L}}$, $A_{k+1} = A_k + \alpha_{k+1}$

3: $\tilde{x}^{k+1} = (A_k x^k + \alpha_{k+1} z^k)/A_{k+1}$

4: $z^{k+1} = z^k - \frac{\alpha_{k+1}}{1+A_{k+1}\mu}\left(\nabla f(\tilde{x}^{k+1}) + \mu\left(z^k - \tilde{x}^{k+1}\right)\right)$

5: $x^{k+1} = (A_k x^k + \alpha_{k+1} z^{k+1})/A_{k+1}$

6: **end for**

Output: x^N

N	μ-strongly convex, L-smooth	L-smooth	μ-strongly convex, $\|\nabla f(x)\|_2 \leq M$	$\|\nabla f(x)\|_2 \leq M$
N # of iterations	$\sqrt{\frac{L}{\mu}} \ln\left(\frac{\mu R^2}{\varepsilon}\right)$	$\sqrt{\frac{LR^2}{\varepsilon}}$	$\frac{M^2}{\mu\varepsilon}$	$\frac{M^2R^2}{\varepsilon^2}$
# of $\nabla f(x)$ oracle calls	$\sqrt{\frac{L}{\mu}} \ln\left(\frac{\mu R^2}{\varepsilon}\right)$	$\sqrt{\frac{LR^2}{\varepsilon}}$	$\frac{M^2}{\mu\varepsilon}$	$\frac{M^2R^2}{\varepsilon^2}$

$L = M^2/(2\delta), \quad \delta = \varepsilon/N$

Optimal algorithm for smooth stochastic convex optimization

Replace $\nabla f(\tilde{x}^{k+1})$ **on**

$$\nabla^{r_{k+1}} f(\tilde{x}^{k+1}, \{\xi_i^{k+1}\}_{i=1}^{r_{k+1}}) = \frac{1}{r_{k+1}} \sum_{i=1}^{r_{k+1}} \nabla f(\tilde{x}^{k+1}, \xi_i^{k+1}),$$

where $\xi_1^{k+1}, \ldots, \xi_{r_{k+1}}^{k+1}$ – i.i.d from the same distribution as ξ and batch size

$$r_{k+1} = O\left(\frac{\sigma^2 \alpha_{k+1} \ln(N/\beta)}{(1 + A_{k+1}\mu)\varepsilon}\right).$$

	μ-strongly convex, L-smooth, $\mathbb{E}\|\nabla f(x,\xi) - \nabla f(x)\|_2^2 \le \sigma^2$	L-smooth, $\mathbb{E}\|\nabla f(x,\xi) - \nabla f(x)\|_2^2 \le \sigma^2$	μ-strongly convex, $\mathbb{E}\|\nabla f(x,\xi)\|_2^2 \le M^2$	$\mathbb{E}\|\nabla f(x,\xi)\|_2^2 \le M^2$
# of iterations	$\sqrt{\frac{L}{\mu}} \ln\left(\frac{\mu R^2}{\varepsilon}\right)$	$\sqrt{\frac{LR^2}{\varepsilon}}$	$\frac{M^2}{\mu\varepsilon}$	$\frac{M^2R^2}{\varepsilon^2}$
# of $\nabla f(x,\xi)$ oracle calls	$\max\left\{\frac{\sigma^2}{\mu\varepsilon}, \sqrt{\frac{L}{\mu}} \ln\left(\frac{\mu R^2}{\varepsilon}\right)\right\}$	$\max\left\{\frac{\sigma^2 R^2}{\varepsilon^2}, \sqrt{\frac{LR^2}{\varepsilon}}\right\}$	$\frac{M^2}{\mu\varepsilon}$	$\frac{M^2R^2}{\varepsilon^2}$

$L = M^2/(2\delta), \quad \delta = \varepsilon/N$

Primal methods for convex optimization with affine constraints

$$f(x) \to \min_{Ax=0, x \in Q},$$

Denote by $R_y = \|y^*\|_2$ 2-norm of the smallest solution y^* of dual (up to a sign) problem

$$R_y^2 \leq \frac{\|\nabla f(x^*)\|_2^2}{\lambda_{\min}^+(A^T A)}.$$

The main trick of this section is to use special penalty method to solve

$$F(x) = f(x) + \frac{R_y^2}{\varepsilon}\|Ax\|_2^2 \to \min_{x \in Q}.$$

Primal methods for convex optimization with affine constraints

The main trick of this section is to use special penalty method to solve

$$F(x) = f(x) + \frac{R_y^2}{\varepsilon}\|Ax\|_2^2 \to \min_{x \in Q}.$$

If

$$F(x^N) - \min_{x \in Q} F(x) \leq \varepsilon,$$

then

$$f(x^N) - \min_{x \in Q} f(x) \leq \varepsilon, \|Ax^N\|_2 \leq \varepsilon / R_y.$$

Accelerated sliding

$$f(x) \to \min_{x \in Q} \qquad g(x) \to \min_{x \in Q}$$

Complexity

$$T_f \text{ calculations } \nabla f(x) \qquad T_g \text{ calculations } \nabla g(x)$$

$$f(x) + g(x) \to \min_{x \in Q}$$

Complexity

$$\tilde{O}\left(T_f\right) \text{ calculations } \nabla f(x) \quad \tilde{O}\left(T_g\right) \text{ calculations } \nabla g(x)$$

Dual methods for convex optimization with affine constraints

$$f(x) \to \min_{Ax=0, x \in Q},$$

The dual problem (up to a sign) is as follows

$$\psi(y) = \varphi(A^T y) = \max_{x \in Q} \{\langle y, Ax \rangle - f(x)\} = \langle y, Ax(A^T y)\rangle - f(x(A^T y)) = \langle A^T y, x(A^T y)\rangle - f(x(A^T y)) \to \min_y .$$

If f is μ-strongly convex in 2-norm, then ψ has $L_\psi = \frac{\lambda_{\max}(A^T A)}{\mu}$–Lipschitz continuous gradient in 2-norm

$$\nabla \psi(y) = Ax(A^T y)$$

$$f(x(A^T y)) - f(x^*) \leq \langle \nabla \psi(y), y \rangle = \langle Ax(A^T y), y \rangle.$$

For

$$f(x^N) - f(x^*) = f(x(A^T y^N)) - f(x(A^T y^*)) \leq 2\varepsilon, \|Ax^N\|_2 \leq \varepsilon / R_y,$$

It is sufficient to find y^N $(\|y^N\|_2 \leq 2R_y)$ **such that** $\|\nabla \psi(y^N)\|_2 \leq \varepsilon / R_y$.

Optimal methods for affine constrained convex problems

DE GRUYTER J. Inverse Ill-Posed Probl. 2021; aop

Research Article

Darina Dvinskikh* and Alexander Gasnikov

Decentralized and parallel primal and dual accelerated methods for stochastic convex programming problems

https://doi.org/10.1515/jiip-2020-0068
Received November 19, 2020; accepted December 10, 2020

Abstract: We introduce primal and dual stochastic gradient oracle methods for decentralized convex optimization problems. Both for primal and dual oracles, the proposed methods are optimal in terms of the number of communication steps. However, for all classes of the objective, the optimality in terms of the number of oracle calls per node takes place only up to a logarithmic factor and the notion of smoothness. By using mini-batching technique, we show that the proposed methods with stochastic oracle can be additionally parallelized at each node. The considered algorithms can be applied to many data science problems and inverse problems.

Keywords: Stochastic optimization, convex optimization, decentralized optimization, complexity bounds, first-order method, mini-batch, sum-type inverse problems

MSC 2010: 90C25, 90C06, 90C15

1 Introduction

Consider the stochastic convex optimization problem

$$\min_{x \in Q \subseteq \mathbb{R}^n} f(x) := \mathbb{E}[f(x, \xi)]. \tag{1.1}$$

Submitted to CDC 2021

Optimal methods for affine constrained convex problems

An Accelerated Method For Decentralized Distributed Stochastic Optimization Over Time-Varying Graphs

Alexander Rogozin[1] aleksandr.rogozin@phystech.edu
Mikhail Bochko[1] bochko.mg@phystech.edu
Pavel Dvurechensky[2] pavel.dvurechensky@wias-berlin.de
Alexander Gasnikov[1] gasnikov@yandex.ru
Vladislav Lukoshkin[3] lukoshkin@phystech.edu

***Abstract*— We consider a distributed stochastic optimization problem that is solved by a decentralized network of agents with only local communication between neighboring agents. The goal of the whole system is to minimize a global objective function given as a sum of local objectives held by each agent. Each local objective is defined as an expectation of a convex smooth random function and the agent is allowed to sample stochastic gradients for this function. For this setting we propose the first accelerated (in the sense of Nesterov's acceleration) method that simultaneously attains optimal up to a logarithmic factor communication and oracle complexity bounds for smooth strongly convex distributed stochastic optimization. We also consider the case when the communication graph is allowed to vary with time and obtain complexity bounds for our algorithm, which are the first upper complexity bounds for this setting in the literature.**

I. Introduction

Distributed algorithms have already about half a century history [1], [2], [3] with many applications including robotics, resource allocation, power system control, control of drone or satellite networks, distributed statistical inference and multiagent reinforcement learning [4], [5], [6], [7], [8], [9], [10]. Recently, development of such algorithms has become one of the main topics in optimization and machine learning motivated by large-scale learning problems with privacy constraints and other challenges such as data being produced or stored distributedly [11], [12], [13], [14], [15]. An important part of this research studies decentralized distributed optimization algorithms over arbitrary networks. In this setting a network of computing agents, e.g. sensors or computers, is represented by a connected graph in which two agents can communicate with each other if there is an edge between them. This imposes communication constraints and the goal of the whole system [16], [17] is to cooperatively

More precisely, we consider the following optimization problem

$$\min_{x\in\mathbb{R}^d} f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x), \quad f_i(x) := \mathbb{E}_{\xi_i\sim\mathcal{D}_i}\mathbf{f}_i(x,\xi_i), \quad (1)$$

where ξ_i's are random variables with probability distributions $\mathcal{D}_i$. For each $i = 1, ..., n$ we make the following assumptions: $f_i(x)$ is a convex function and that almost sure w.r.t. distribution $\mathcal{D}_i$, the function $\mathbf{f}_i(x,\xi_i)$ has gradient $\nabla\mathbf{f}_i(x,\xi_i)$, which is $L_i(\xi)$-Lipschitz continuous with respect to the Euclidean norm. Further, for each $i = 1, ..., n$, we assume that we know a constant $L_i \geqslant 0$ such that $\sqrt{\mathbb{E}_\xi L_i(\xi)^2} \leq L_i < +\infty$. Under these assumptions, $\mathbb{E}_\xi\nabla\mathbf{f}_i(x,\xi_i) = \nabla f_i(x)$ and f is L_i-smooth, i.e. has L_i-Lipschitz continuous gradient with respect to the Euclidean norm. Also we assume that, for all x, and i

$$\mathbb{E}_\xi[\|\nabla\mathbf{f}_i(x,\xi_i) - \nabla f_i(x)\|^2] \leqslant \sigma_i^2, \quad (2)$$

where $\|\cdot\|$ is the Euclidean norm. We also assume that each f_i is $\mu_i > 0$-strongly convex. Important characteristics of the objective in (1) are local strong convexity parameter $\mu_l = \min_i \mu_i$ and local smoothness constant $L_l = \max_i L_i$, which define local condition number $\kappa_l = L_l/\mu_l$, as well as their global counterparts $\mu_g = \frac{1}{n}\sum_{i=1}^n \mu_i$, $L_g = \frac{1}{n}\sum_{i=1}^n L_i$, $\kappa_g = L_g/\mu_g$. The global conditioning may be significantly better than local (see e.g. [18] for details) and it is desired to develop algorithms with complexity depending on the global condition number. Moreover, we introduce a worst-case smoothness constant over stochastic realizations $L_\xi = \max_i \max_\xi L_i(\xi)$ and a maximum gradient norm at optimum $M_\xi = \max_i \max_\xi \|\nabla f_i(x^*,\xi)\|$ and assume that these con-

Sum type target function

$$f(x) = \mathbb{E}[f(x, \xi)] \to \min_{x \in Q \subseteq \mathbb{R}^n}.$$

How to choose m in:

$$\min_{x \in Q \subseteq \mathbb{R}^n} \frac{1}{m} \sum_{k=1}^{m} f(x, \xi^k) + \frac{\varepsilon}{2R^2} \|x - x^0\|_2^2$$

$\|x^0 - x_*\|_2$

Answer (up to a log-factor):

$$m = \min \left\{ O\left(\frac{M^2 R^2}{\varepsilon^2}\right), O\left(\frac{M^2}{\mu\varepsilon}\right) \right\}$$

Where:

Strong convexity constant of f

$$\mathbb{E}[\|\nabla f(x, \xi)\|_2^2] \leq M^2$$

S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. Stochastic convex optimization. In *COLT*, 2009.

Decentralized distributed optimization

$$f(x) = \frac{1}{m}\sum_{k=1}^{m} f_k(x) \to \min_{x \in Q \subseteq \mathbb{R}^n}.$$

$$\bar{W}_{ij} = \begin{cases} -1, & \text{if } (i,j) \in E, \\ \deg(i), & \text{if } i = j, \\ 0, & \text{otherwise,} \end{cases} \qquad W = \bar{W} \otimes I_n$$

$$x_1 = ... = x_m \in \mathbb{R}^n \text{ as } W\mathbf{x} = 0, \text{ moreover, as } \sqrt{W}\mathbf{x} = 0$$

$$F(\mathbf{x}) = \frac{1}{m}\sum_{k=1}^{m} f_k(x_k) \to \min_{\substack{x_1 = \cdots = x_m, \\ x_1, \ldots, x_m \in Q \subseteq \mathbb{R}^n}}. \qquad F(\mathbf{x}) = \frac{1}{m}\sum_{k=1}^{m} f_k(x_k) \to \min_{\substack{\sqrt{W}\mathbf{x} = 0, \\ x_1, \ldots, x_m \in Q \subseteq \mathbb{R}^n}}$$

Decentralized distributed optimization

$$f(x) = \frac{1}{m}\sum_{k=1}^{m} f_k(x) \to \min_{x \in Q \subseteq \mathbb{R}^n}.$$

$$F(\mathbf{x}) = \frac{1}{m}\sum_{k=1}^{m} f_k(x_k) \to \min_{\substack{\sqrt{W}\mathbf{x}=0,\\ x_1,\ldots,x_m \in Q \subseteq \mathbb{R}^n}}$$

all f_k are M-Lipschitz, L-smooth and μ-strongly convex (it is possible that, $L = \infty$ or (and) $\mu = 0$).

$$\mathbb{E}\left[\exp\left(\frac{\|\nabla f_k(x_k, \xi_k) - \nabla f_k(x_k)]\|_2^2}{\sigma^2}\right)\right] \leq \exp(1).$$

Dual decentralized distributed optimization

$$F(\mathbf{x}) = \frac{1}{m}\sum_{k=1}^{m} f_k(x_k) \to \min_{\substack{\sqrt{W}\mathbf{x}=0,\\ x_1,\ldots,x_m \in Q \subseteq \mathbb{R}^n}}$$

$$A = \sqrt{W},\ L_F = \frac{L}{m},\ \mu_F = \frac{\mu}{m},\ \|\nabla F(\mathbf{x})\|_2^2 \le M_F^2 = \frac{M^2}{m},\ \sigma_F^2 = O\left(\frac{\sigma^2}{m}\right),$$

$$R_{\mathbf{x}}^2 = \|\mathbf{x}^0 - \mathbf{x}^*\|_2^2 = m\|x^0 - x^*\|_2^2 = mR^2,\ R_{\mathbf{y}}^2 = \|\mathbf{y}^*\|_2^2 \le \frac{\|\nabla F(\mathbf{x}^*)\|_2^2}{\lambda_{\min}^+(W)} \le \frac{M^2}{m\lambda_{\min}^+(W)}.$$

$A^T A x = \sqrt{W}^T \sqrt{W} x = W x$ **– calculated in a decentralized distributed manner!**

$$\chi = \frac{\lambda_{\max}(W)}{\lambda_{\min}^+(W)}.$$

Decentralized distributed optimization

$$F(\mathbf{x}) = \frac{1}{m}\sum_{k=1}^{m} f_k(x_k) \to \min_{\substack{\sqrt{W}\mathbf{x}=0,\\ x_1,\ldots,x_m \in Q \subseteq \mathbb{R}^n}}$$

Dual approach

$$\Psi(\mathbf{y}) = \frac{1}{m}\Phi(m\sqrt{W}\mathbf{y}) = \frac{1}{m}\sum_{k=1}^{m} \varphi_k(m[\sqrt{W}\mathbf{y}]_k) \to \min_{\mathbf{y}\in\mathbb{R}^{mn}},$$

$$\varphi_k(\lambda_k) = \max_{x_k \in Q \subseteq \mathbb{R}^n} \{\langle \lambda_k, x_k\rangle - f_k(x_k)\}$$

$$\varphi_k(\lambda_k) \;=\; \mathbb{E}[\varphi_k(\lambda_k, \xi_k)].$$

Decentralized distributed optimization

$$\Psi(\mathbf{y}) = \frac{1}{m}\Phi(m\sqrt{W}\mathbf{y}) = \frac{1}{m}\sum_{k=1}^{m}\varphi_k(m[\sqrt{W}\mathbf{y}]_k) \to \min_{\mathbf{y}\in\mathbb{R}^{mn}},$$

$$\varphi_k(\lambda_k) = \max_{x_k\in Q\subseteq\mathbb{R}^n}\{\langle\lambda_k, x_k\rangle - f_k(x_k)\}$$

$$\varphi_k(\lambda_k) = \mathbb{E}[\varphi_k(\lambda_k, \xi_k)].$$

$$\mathbb{E}\left[\exp\left(\frac{\|\nabla\varphi_k(\lambda_k,\xi_k) - \nabla\varphi_k(\lambda_k)]\|_2^2}{\sigma_\varphi^2}\right)\right] \leq \exp(1).$$

$$\boxed{A = \sqrt{W},\ \sigma_\Psi^2 = O\left(\lambda_{\max}(W)m\sigma_\varphi^2\right)\ .}$$

Since $x(A^Ty) = \mathbf{x}(\sqrt{W}\mathbf{y})$ we should change the variables: $\tilde{\mathbf{y}} := \sqrt{W}\tilde{\mathbf{y}}$, $\mathbf{z} := \sqrt{W}\mathbf{z}$, $\mathbf{y} := \sqrt{W}\mathbf{y}$.

Decentralized distributed optimization
Optimal bounds for primal oracle

	f_k is μ-strongly convex, and L-smooth	f_k is L-smooth	f_k is μ-strongly convex	
# of communication rounds	$\widetilde{O}\left(\sqrt{\frac{L}{\mu}}\chi\right)$	$\widetilde{O}\left(\sqrt{\frac{LR^2}{\varepsilon}}\chi\right)$	$O\left(\sqrt{\frac{M^2}{\mu\varepsilon}}\chi\right)$	$O\left(\sqrt{\frac{M^2R^2}{\varepsilon^2}}\chi\right)$
# of $\nabla f_k(x_k)$ oracle calls per node k	$\widetilde{O}\left(\sqrt{\frac{L}{\mu}}\right)$	$O\left(\sqrt{\frac{LR^2}{\varepsilon}}\right)$	$O\left(\frac{M^2}{\mu\varepsilon}\right)$	$O\left(\frac{M^2R^2}{\varepsilon^2}\right)$
Algorithm	PSTM, $Q=\mathbb{R}^n$	PSTM, $Q=\mathbb{R}^n$	R-Sliding	Sliding

NEW! arXiv:2103.15598

	f_k is μ-strongly convex and L-smooth	f_k is L-smooth	f_k is μ-strongly convex,	
# of communication rounds	$\widetilde{O}\left(\sqrt{\frac{L}{\mu}}\chi\right)$	$\widetilde{O}\left(\sqrt{\frac{LR^2}{\varepsilon}}\chi\right)$	$O\left(\sqrt{\frac{M^2}{\mu\varepsilon}}\chi\right)$	$O\left(\sqrt{\frac{M^2R^2}{\varepsilon^2}}\chi\right)$
# of $\nabla f_k(x_k,\xi_k)$ oracle calls per node k	$\widetilde{O}\left(\max\left\{\frac{\sigma^2}{m\mu\varepsilon}, \sqrt{\frac{L}{\mu}}\right\}\right)$	$O\left(\max\left\{\frac{\sigma^2R^2}{m\varepsilon^2}, \sqrt{\frac{LR^2}{\varepsilon}}\right\}\right)$	$O\left(\frac{M^2+\sigma^2}{\mu\varepsilon}\right)$	$O\left(\frac{(M^2+\sigma^2)R^2}{\varepsilon^2}\right)$
Algorithm	PBSTM, $Q=\mathbb{R}^n$	PBSTM, $Q=\mathbb{R}^n$	Stochastic R-Sliding	Stochastic Sliding

Decentralized distributed optimization
Optimal bounds for dual oracle

NEW!

[arXiv:2011.13259](#)	f_k is μ-strongly convex, and L-smooth	f_k is μ-strongly convex
# of communication rounds	$\widetilde{O}\left(\sqrt{\frac{L}{\mu}}\chi\right)$	$O\left(\sqrt{\frac{M^2}{\mu\varepsilon}}\chi\right)$
# of $\nabla\varphi_k(\lambda_k)$ oracle calls per node k	$\widetilde{O}\left(\sqrt{\frac{L}{\mu}}\chi\right)$	$O\left(\sqrt{\frac{M^2}{\mu\varepsilon}}\chi\right)$
Algorithm	ROGM-G or STM, $Q = \mathbb{R}^n$	OGM-G or PDSTM

[arXiv:2011.13259](#)	f_k is μ-strongly convex, and L-smooth	f_k is μ-strongly convex
# of communication rounds	$\widetilde{O}\left(\sqrt{\frac{L}{\mu}}\chi\right)$	$O\left(\sqrt{\frac{M^2}{\mu\varepsilon}}\chi\right)$
# of $\nabla\varphi_k(\lambda_k, \xi_k)$ oracle calls per node k	$\widetilde{O}\left(\max\left\{\frac{M^2\sigma_\phi^2}{\varepsilon^2}\chi, \sqrt{\frac{L}{\mu}}\chi\right\}\right)$	$O\left(\max\left\{\frac{M^2\sigma_\phi^2}{\varepsilon^2}\chi, \sqrt{\frac{M^2}{\mu\varepsilon}}\chi\right\}\right)$
Algorithm	R-RRMA+AC-SA^2, $Q = \mathbb{R}^n$	SPDSTM

NEW!

Wasserstein distance and dual representation

$$U(\mu,\nu) \triangleq \{\pi \in \mathbb{R}_+^{n_2 \times n_1} : \pi \mathbf{1}_{n_1} = \mu, \pi^T \mathbf{1}_{n_2} = \nu\}.$$

$$\min_{\pi \in U(\mu,\nu)} \langle C, \pi \rangle.$$

$$W_\gamma(p,q) = \min_{\pi \in U(p,q)} \{\langle C, \pi \rangle - \gamma \langle \pi, \log \pi \rangle\}$$

$$\begin{aligned} W^*_{\gamma,q}(u) &= \max_{p \in \Delta_n} \{\langle u, p \rangle - W_\gamma(p,q)\} \\ &= \gamma \left(-\langle q, \log q \rangle + \sum_{j=1}^n [q]_j \log \left(\sum_{i=1}^n \exp\left(([u]_i - C_{ji})/\gamma\right) \right) \right), \end{aligned} \tag{1.5}$$

$$\forall l = 1, ..., n \qquad [\nabla W^*_{\gamma,q}(u)]_l = \sum_{j=1}^n [q]_j \frac{\exp\left(([u]_l - C_{lj})/\gamma\right)}{\sum_{\ell=1}^n \exp\left(([u]_\ell - C_{\ell j})/\gamma\right)}. \tag{1.6}$$

Wasserstein barycenter

Decentralized formulation of the Wasserstein barycenter problem both for the saddle-point and dual representation is based on introducing artificial constraint $p_1 = p_2 = ... = p_m$ which is further replaced with affine constraint $\mathbf{W}\mathbf{p} = 0$ (in the saddle-point approach) and $\sqrt{\mathbf{W}}\mathbf{p} = 0$ (in the dual approach), where $\mathbf{p} = (p_1^\top, ..., p_m^\top)^\top$ is column vector and $\mathbf{W}$ is known as communication matrix for a decentralized system. From the definition of matrix $\mathbf{W}$ it follows that

$$\sqrt{\mathbf{W}}\mathbf{p} = 0, \text{ and } \mathbf{W}\mathbf{p} = 0 \Longleftrightarrow p_1 = p_2 = ... = p_m.$$

Affine constraint $\mathbf{W}\mathbf{p} = 0$ (or $\sqrt{\mathbf{W}}\mathbf{p} = 0$) is brought to the objective via the Fenchel–Legendre transform. For the saddle-point problem (primal approach) the structure of the problem preserves. For dual approach, the Fenchel–Legendre transform leads to the dual functions (1.5) with closed-form representations. The primal Wasserstein barycenter problem defined w.r.t. entropy-regularized optimal transport is formulated as follows

$$\min_{p \in \Delta_n} \frac{1}{m} \sum_{i=1}^{m} W_\gamma(p, q_i) = \min_{\substack{p_1 = ... = p_m, \\ p_1, ..., p_m \in \Delta_n}} \frac{1}{m} \sum_{i=1}^{m} W_\gamma(p_i, q_i) = \min_{\substack{\sqrt{\mathbf{W}}\mathbf{p} = 0, \\ p_1, ..., p_m \in \Delta_n}} \frac{1}{m} \sum_{i=1}^{m} W_\gamma(p_i, q_i).$$

Wasserstein barycenter and dual representation

$$\min_{\mathbf{y}\in\mathbb{R}^{nm}} W^*_{\gamma,\mathbf{q}}(\sqrt{\mathbf{W}}\mathbf{y}) \triangleq \frac{1}{m}\sum_{i=1}^{m} W^*_{\gamma,q_i}([\sqrt{\mathbf{W}}\mathbf{y}]_i), \tag{1.7}$$

where $\mathbf{y} = (y_1^\top \cdots y_m^\top)^\top \in \mathbb{R}^{nm}$ is the Lagrangian dual multiplier. The decentralized procedure solving (1.7) can be demostrated on the gradient descent for L-Lipschitz smooth function (the constant L for $W^*_{\gamma,\mathbf{q}}(\sqrt{\mathbf{W}}\mathbf{y})$ is defined via $\mathbf{W}$ and regularized parameter γ)

$$\mathbf{y}^{k+1} = \mathbf{y}^k - \frac{1}{L}\nabla W^*_{\gamma,\mathbf{q}}(\sqrt{\mathbf{W}}\mathbf{y}^k) = \mathbf{y}^k - \frac{1}{L}\sqrt{\mathbf{W}}\mathbf{p}(\sqrt{\mathbf{W}}\mathbf{y}^k).$$

Without change of variable, it is unclear how to execute this procedure in a distributed fashion. Let $\mathbf{u} := \sqrt{W}\mathbf{y}$, then the gradient step multiplied by $\sqrt{\mathbf{W}}$ can be rewritten as

$$\mathbf{u}^{k+1} = \mathbf{u}^k - \frac{1}{L}\mathbf{W}\mathbf{p}(\mathbf{u}^k),$$

where $[\mathbf{p}(\boldsymbol{u})]_i = p_i(u_i) = \nabla W^*_{\gamma,q_i}(u_i)$ (1.6), $i = 1, ..., m$ The procedure can be performed in a decentralized manner on a network of agents. Vector $\mathbf{W}\mathbf{p}(\boldsymbol{u})$ naturally defines communications with neighboring nodes due to the structure of communication matrix since the elements of communication matrix are zero for non-neighboring nodes.

WB dual stochastic oracle

$$\forall l = 1, ..., n \qquad [\nabla W^*_{\gamma,q}(u)]_l = \sum_{j=1}^{n} [q]_j \frac{\exp\left(([u]_l - C_{lj})/\gamma\right)}{\sum_{\ell=1}^{n} \exp\left(([u]_\ell - C_{\ell j})/\gamma\right)}. \tag{1.6}$$

Moreover, in the dual approach which is based on gradient method, the randomization of $\nabla W^*_{\gamma,q_i}$ can be used (stochastic approximation of $\nabla W^*_{\gamma,q_i}$) to reduce the complexity of calculating the true gradient, that is $O(n^2)$, by calculating its stochastic approximation of $O(n)$ complexity. The randomization for the true gradient (1.6) is achieved by taking the j-th term in the sum with probability $[q]_j$

$$[\nabla W^*_{\gamma,q}(u,\xi)]_l = \frac{\exp\left(([u]_l - C_{l\xi})/\gamma\right)}{\sum_{\ell=1}^{n} \exp\left(([u]_\ell - C_{\ell\xi})/\gamma\right)}, \qquad \forall l = 1, ..., n.$$

where we replaced index j by ξ to underline its randomness. This is the motivation for considering the first-order methods with stochastic oracle.

Saddle-point generalizations

arXiv:2102.07758 [pdf, other] math.OC cs.DC

Decentralized Distributed Optimization for Saddle Point Problems

Authors: Alexander Rogozin, Alexander Beznosikov, Darina Dvinskikh, Dmitry Kovalev, Pavel Dvurechensky, Alexander Gasnikov

Abstract: We consider distributed convex-concave saddle point problems over arbitrary connected undirected networks and propose a decentralized distributed algorithm for their solution. The local functions distributed across the nodes are assumed to have global and local groups of variables. For the proposed algorithm we prove non-asymptotic convergence rate estimates with explicit dependence on the network... ▽ More

Submitted 16 February, 2021; **v1** submitted 15 February, 2021; **originally announced** February 2021.

arXiv:2010.13112 [pdf, other] cs.LG cs.DC math.OC

Distributed Saddle-Point Problems: Lower Bounds, Optimal Algorithms and Federated GANs

Authors: Aleksandr Beznosikov, Valentin Samokhin, Alexander Gasnikov

Abstract: GAN is one of the most popular and commonly used neural network models. When the model is large and there is a lot of data, the learning process can be delayed. The standard way out is to use multiple devices. Therefore, the methods of distributed and federated training for GANs are an important question. But from an optimization point of view, GANs are saddle-point problems:... ▽ More

Submitted 27 February, 2021; **v1** submitted 25 October, 2020; **originally announced** October 2020.

Time-varying networks

arXiv:2103.15598 [pdf, other] math.OC

An Accelerated Method For Decentralized Distributed Stochastic Optimization Over Time-Varying Graphs

Authors: Alexander Rogozin, Mikhail Bochko, Pavel Dvurechensky, Alexander Gasnikov, Vladislav Lukoshkin

Abstract: We consider a distributed stochastic optimization problem that is solved by a decentralized network of agents with only local communication between neighboring agents. The goal of the whole system is to minimize a global objective function given as a sum of local objectives held by each agent. Each local objective is defined as an expectation of a convex smooth random function and the agent is all... ▽ More

Submitted 29 March, 2021; **originally announced** March 2021.

arXiv:2102.09234 [pdf, other] math.OC cs.LG

ADOM: Accelerated Decentralized Optimization Method for Time-Varying Networks

Authors: Dmitry Kovalev, Egor Shulgin, Peter Richtárik, Alexander Rogozin, Alexander Gasnikov

Abstract: We propose ADOM - an accelerated method for smooth and strongly convex decentralized optimization over time-varying networks. ADOM uses a dual oracle, i.e., we assume access to the gradient of the Fenchel conjugate of the individual loss functions. Up to a constant factor, which depends on the network structure only, its communication complexity is the same as that of accelerated Nesterov gradient... ▽ More

Submitted 18 February, 2021; **originally announced** February 2021.

Time-varying networks

arXiv:2102.07758 [pdf, other] math.OC cs.DC

Decentralized Distributed Optimization for Saddle Point Problems

Authors: Alexander Rogozin, Alexander Beznosikov, Darina Dvinskikh, Dmitry Kovalev, Pavel Dvurechensky, Alexander Gasnikov

Abstract: We consider distributed convex-concave saddle point problems over arbitrary connected undirected networks and propose a decentralized distributed algorithm for their solution. The local functions distributed across the nodes are assumed to have global and local groups of variables. For the proposed algorithm we prove non-asymptotic convergence rate estimates with explicit dependence on the network... ▽ More

Submitted 16 February, 2021; **v1** submitted 15 February, 2021; **originally announced** February 2021.

arXiv:2009.11069 [pdf, other] math.OC

Towards accelerated rates for distributed optimization over time-varying networks

Authors: Alexander Rogozin, Vladislav Lukoshkin, Alexander Gasnikov, Dmitry Kovalev, Egor Shulgin

Abstract: We study the problem of decentralized optimization over time-varying networks with strongly convex smooth cost functions. In our approach, nodes run a multi-step gossip procedure after making each gradient update, thus ensuring approximate consensus at each iteration, while the outer loop is based on accelerated Nesterov scheme. The algorithm achieves precision $\varepsilon > 0$ in... ▽ More

Submitted 15 January, 2021; **v1** submitted 23 September, 2020; **originally announced** September 2020.

Comments
Centralized vs Decentralized optimization

First difference: is a replacement

$$d \text{ on } \tilde{O}(\sqrt{\chi})$$

in number of communications steps, where d is a diameter of graph

Second difference: is a replacement of average constants L, μ (used in Centralized approach) for worth ones (used in Decentralized approach). Fortunately, the this problem was partially solved for primal oracle:

arXiv:2009.11069 [pdf, other] math.OC

Towards accelerated rates for distributed optimization over time-varying networks

Authors: Alexander Rogozin, Vladislav Lukoshkin, Alexander Gasnikov, Dmitry Kovalev, Egor Shulgin

Abstract: We study the problem of decentralized optimization over time-varying networks with strongly convex smooth cost functions. In our approach, nodes run a multi-step gossip procedure after making each gradient update, thus ensuring approximate consensus at each iteration, while the outer loop is based on accelerated Nesterov scheme. The algorithm achieves precision $\varepsilon > 0$ in... ▽ More

Submitted 15 January, 2021; **v1** submitted 23 September, 2020; **originally announced** September 2020.

arXiv:2103.15598 [pdf, other] math.OC

An Accelerated Method For Decentralized Distributed Stochastic Optimization Over Time-Varying Graphs

Authors: Alexander Rogozin, Mikhail Bochko, Pavel Dvurechensky, Alexander Gasnikov, Vladislav Lukoshkin

Abstract: We consider a distributed stochastic optimization problem that is solved by a decentralized network of agents with only local communication between neighboring agents. The goal of the whole system is to minimize a global objective function given as a sum of local objectives held by each agent. Each local objective is defined as an expectation of a convex smooth random function and the agent is all... ▽ More

Submitted 29 March, 2021; **originally announced** March 2021.