

Современная стохастическая оптимизация и анализ данных

Гасников А.В. (МФТИ)

gasnikov.av@mipt.ru

Based on joint work with Dmitry Kovalev, Alexander Beznosikov, Ekaterina Borodich and Aldo Scutari.

AISTATS2022, NeurIPS 2022

[arXiv:2205.15136](https://arxiv.org/abs/2205.15136)

Вторая конференция Математических центров России; November 7, 2022

Structure of the Lecture

1. Preliminaries from Machine Learning and Convex optimization

2. Monte Carlo approach and sum-type optimization problems

3. Variance reduction (VR) vs Statistical Similarity

4. Gradient descent with relative smoothness and strong convexity and relation with Similarity setup

5. Lower bound for Similarity and the problem of acceleration

6. Optimal algorithm

7. Sum-type Saddle-point problems (SPP) - no need of acceleration. Optimal methods

Sum type target convex function

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

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:

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

Strong convexity constant of f

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

Too many terms!

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

We skip this term
for simplicity

Problem: How to store data in memory?

Answer: To use distributed approach

Problem: Too many communications and oracle calls required

Answer: To store at each node a lot of data. And rewrite optimization problem

$$\min_{x \in Q} F(x) := \frac{1}{M} \sum_{k=1}^M \left(\frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i}) \right)$$
$$F_k(x) = \frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i})$$

This problem has specific structure

1) $F_k(x)$ has sum-type structure and variance reduction is possible

2) Statistical Similarity $\|\nabla^2 F_k(x) - \nabla^2 F(x)\| = O\left(\sqrt{\frac{L_2^2 n}{r}}\right)$

Variance reduction

$$\min_{x \in Q} F(x) := \frac{1}{M} \sum_{k=1}^M \left(\frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i}) \right)$$

This problem has specific structure

$$F_k(x) = \frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i})$$

1) $F_k(x)$ has sum-type structure and variance reduction is possible

Variance Reduced EXTRA and DIGing and Their Optimal Acceleration for Strongly Convex Decentralized Optimization

Huan Li¹ Zhouchen Lin² Yongchun Fang¹

Abstract

We study stochastic decentralized optimization for the problem of training machine learning models with large-scale distributed data. We extend the widely used EXTRA and DIGing methods with variance reduction (VR), and propose two methods: VR-EXTRA and VR-DIGing. The proposed VR-EXTRA requires the time of $\mathcal{O}((\kappa_s + n) \log \frac{1}{\epsilon})$ stochastic gradient evaluations and $\mathcal{O}((\kappa_b + \kappa_c) \log \frac{1}{\epsilon})$ communication rounds to reach precision ϵ , which are the best complexities among the non-accelerated gradient-type methods, where κ_s and κ_b are the stochastic condition number and batch condition number for strongly convex and smooth problems, respectively, κ_c is the condition number of the communication network, and n is the sample size on each distributed node. The proposed VR-DIGing has a little higher communication cost of $\mathcal{O}((\kappa_b + \kappa_c^2) \log \frac{1}{\epsilon})$. Our stochastic gradient computation complexities are the same as the ones of single-machine VR methods, such as SAG, SAGA, and SVRG, and our communication complexities keep the same as those of EXTRA and DIGing, respectively. To further speed up the convergence, we also propose the accelerated VR-EXTRA and VR-DIGing with both the optimal $\mathcal{O}((\sqrt{n\kappa_s} + n) \log \frac{1}{\epsilon})$ stochastic gradient computation complexity and $\mathcal{O}(\sqrt{\kappa_b \kappa_c} \log \frac{1}{\epsilon})$ communication complexity. Our stochastic gradient computation complexity is also the same as the ones of single-machine accelerated VR methods, such as Katyusha, and our communication complexity keeps the same as those of accelerated full batch decentralized methods, such as MSDA. To the best of our knowledge, our accelerated methods are the first to achieve both the optimal stochastic gradient computation complexity and communication complexity in the class of gradient-type methods.

Variance reduction

$$\min_{x \in Q} F(x) := \frac{1}{M} \sum_{k=1}^M \left(\frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i}) \right)$$

This problem has specific structure 

1) $F_k(x)$ has sum-type structure and variance reduction is possible

Assumptions:

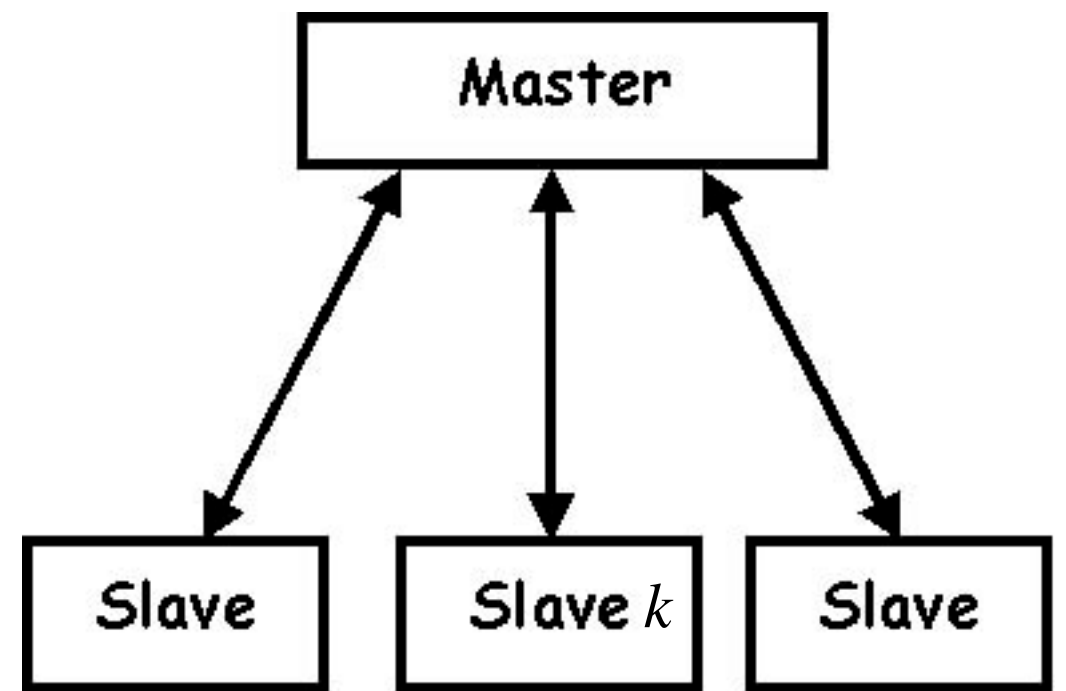
$f(x, \xi)$ is L -smooth in x for all ξ


$f(x, \xi)$ is λ -strongly convex in x for all ξ

Optimal bounds for general $f_{k,i}(x) \neq f(x, \xi^{k,i})$:

$$O\left(\sqrt{\frac{L}{\lambda}} \log\left(\frac{LR^2}{\varepsilon}\right)\right) \quad \text{Communication rounds}$$

$$O\left(\left(r + \sqrt{r \frac{L}{\lambda}}\right) \log\left(\frac{LR^2}{\varepsilon}\right)\right) \quad \text{Oracle calls per node}$$



$$F_k(x) = \frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i})$$


Statistical Similarity

$$\min_{x \in Q} F(x) := \frac{1}{M} \sum_{k=1}^M \left(\frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i}) \right)$$

This problem has specific structure 

1) $F_k(x)$ has sum-type structure and variance reduction is possible

Assumptions:

$f(x, \xi)$ is L -smooth in x for all ξ

$f(x, \xi)$ is λ -strongly convex in x for all ξ

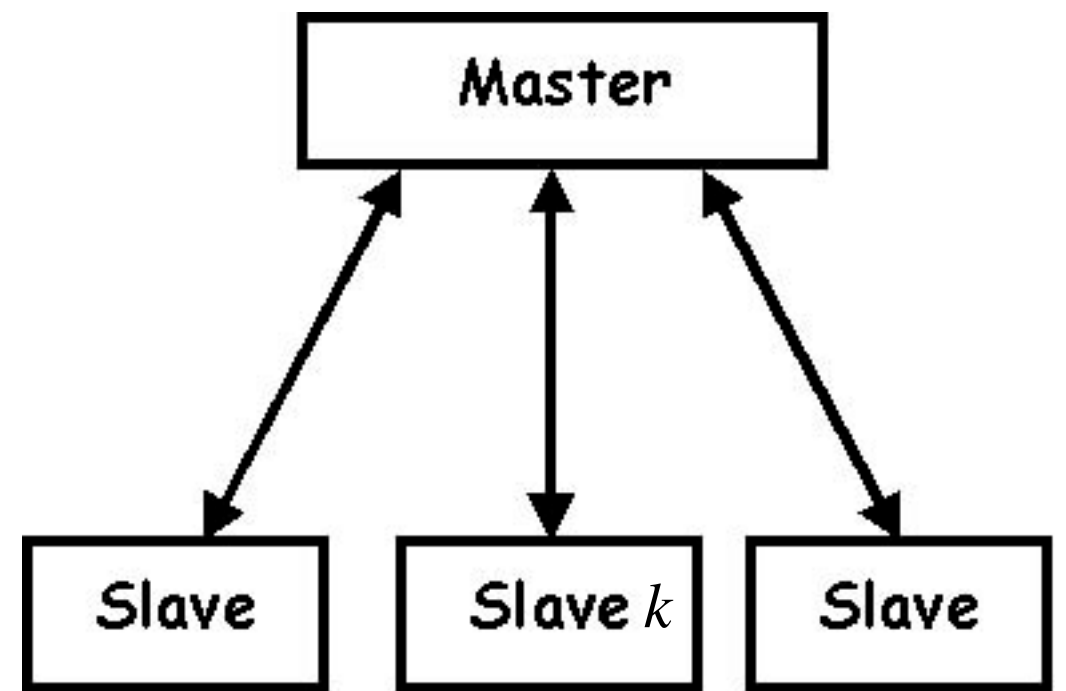
Optimal bound ?:

$$O \left(\sqrt{\frac{L}{\lambda}} \log \left(\frac{LR^2}{\varepsilon} \right) \right) \quad \text{Communication rounds}$$



Is this bound optimal?

Answer: No, if we use similarity: $f_{k,i}(x) = f(x, \xi^{k,i})$, $\xi^{k,i}$ i.i.d.



$$F_k(x) = \frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i})$$

Statistical Similarity

$$\min_{x \in Q} F(x) := \frac{1}{M} \sum_{k=1}^M \left(\frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i}) \right)$$

This problem has specific structure

$$F_k(x) = \frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i})$$

1) $F_k(x)$ has sum-type structure and variance reduction is possible

Assumptions:

$f(x, \xi)$ is L -smooth in x for all $\xi \Rightarrow \|\nabla^2 F_k(x)\| \leq L$

$f(x, \xi)$ is λ -strongly convex in x for all ξ

Communication rounds

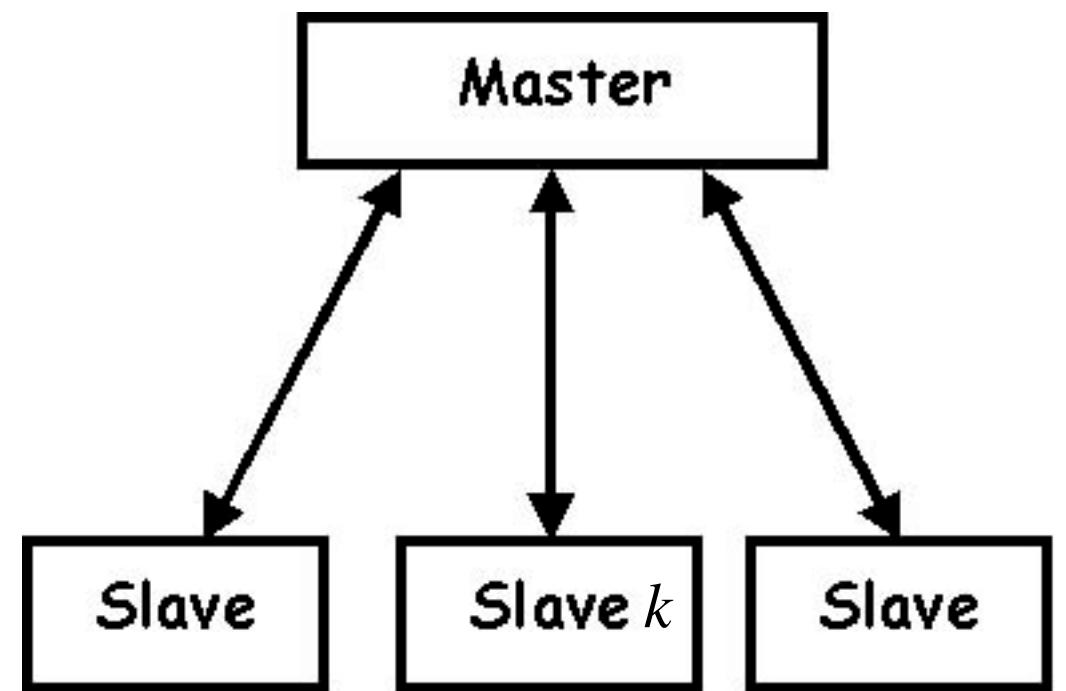
$$O\left(\sqrt{\frac{L}{\lambda}} \log\left(\frac{LR^2}{\varepsilon}\right)\right)$$

Variance reduction

$$O\left(\sqrt{\frac{\delta}{\lambda}} \log\left(\frac{LR^2}{\varepsilon}\right)\right)$$

$$\|\nabla^2 F_k(x) - \nabla^2 F(x)\| \leq \delta$$

δ -Similarity



$$F_k(x) = \frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i})$$

Gradient method with relative smoothness and strong convexity

$$\min_{x \in Q} F(x) := \frac{1}{M} \sum_{k=1}^M F_k(x)$$

$d(x)$ - Smooth convex function

$V(y, x) = d(y) - d(x) - \langle \nabla d(x), y - x \rangle$ - Bregman divergence

SIAM J. OPTIM.
Vol. 28, No. 1, pp. 333–354

© 2018 Society for Industrial and Applied Mathematics

RELATIVELY SMOOTH CONVEX OPTIMIZATION BY FIRST-ORDER METHODS, AND APPLICATIONS*

HAIHAO LU[†], ROBERT M. FREUND[‡], AND YURII NESTEROV[§]

Abstract. The usual approach to developing and analyzing first-order methods for smooth convex optimization assumes that the gradient of the objective function is uniformly smooth with some Lipschitz constant L . However, in many settings the differentiable convex function $f(\cdot)$ is not uniformly smooth—for example, in D -optimal design where $f(x) := -\ln \det(HXH^T)$ and $X := \text{Diag}(x)$, or even the univariate setting with $f(x) := -\ln(x) + x^2$. In this paper we develop a notion of “relative smoothness” and relative strong convexity that is determined relative to a user-specified “reference function” $h(\cdot)$ (that should be computationally tractable for algorithms), and we show that many differentiable convex functions are relatively smooth with respect to a correspondingly fairly simple reference function $h(\cdot)$. We extend two standard algorithms—the primal gradient scheme and the dual averaging scheme—to our new setting, with associated computational guarantees. We apply our new approach to develop a new first-order method for the D -optimal design problem, with associated computational complexity analysis. Some of our results have a certain overlap with the recent work [H. H. Bauschke, J. Bolte, and M. Teboulle, *Math. Oper. Res.*, 42 (2017), pp. 330–348].

Gradient method with relative smoothness and strong convexity

$$\min_{x \in Q} F(x) := \frac{1}{M} \sum_{k=1}^M F_k(x)$$

$$\lambda \nabla^2 d(x) \prec \nabla^2 F(x) \prec L \nabla^2 d(x)$$

$$V(y, x) = d(y) - d(x) - \langle \nabla d(x), y - x \rangle$$

$$x^{k+1} = \arg \min_{x \in Q} \left\{ F(x^k) + \langle \nabla F(x^k), x - x^k \rangle + L V(x, x^k) \right\}$$

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

$$N = O \left(\frac{L}{\lambda} \log \left(\frac{\Delta F}{\varepsilon} \right) \right)$$

Gradient method with relative smoothness and strong convexity and Similarity

$$\min_{x \in Q} F(x) := \frac{1}{M} \sum_{k=1}^M F_k(x)$$

$$d(x) = F_1(x) + \frac{\delta}{2} \|x\|^2$$

Available at master
node (1)

$$\frac{\lambda}{\lambda + 2\delta} \nabla^2 d(x) \prec \nabla^2 F(x) \prec \nabla^2 d(x)$$

$$x^{k+1} = \arg \min_{x \in Q} \left\{ F(x^k) + \langle \nabla F(x^k), x - x^k \rangle + V(x, x^k) \right\}$$

Available at master
node (1) via
communications

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

$$N = O \left(\frac{\max\{\delta, \lambda\}}{\lambda} \log \left(\frac{\Delta F}{\varepsilon} \right) \right)$$

Gradient method with relative smoothness and strong convexity and Similarity

$$N = O \left(\frac{\max\{\delta, \lambda\}}{\lambda} \log \left(\frac{\Delta F}{\varepsilon} \right) \right) = O \left(\frac{\delta}{\lambda} \log \left(\frac{\Delta F}{\varepsilon} \right) \right)$$

Warning: Unfortunately, this rate is not optimal (accelerated)!

But! Accelerated method with relative smoothness and strong convexity is in principle impossible in general set up.

Full Length Paper | [Published: 21 April 2021](#)

Optimal complexity and certification of Bregman first-order methods

[Radu-Alexandru Dragomir](#) , [Adrien B. Taylor](#), [Alexandre d'Aspremont](#) & [Jérôme Bolte](#)

[Mathematical Programming](#) (2021) | [Cite this article](#)

169 Accesses | **0** Altmetric | [Metrics](#)

Statistical Similarity

$$\min_{x \in Q} F(x) := \frac{1}{M} \sum_{k=1}^M \left(\frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i}) \right)$$

This problem has specific structure

$$F_k(x) = \frac{1}{r} \sum_{i=1}^r f(x, \xi^{k,i})$$

1) $F_k(x)$ has sum-type structure and variance reduction is possible

Assumptions:

$f(x, \xi)$ is L -smooth in x for all $\xi \Rightarrow \|\nabla^2 F_k(x)\| \leq L$

$f(x, \xi)$ is λ -strongly convex

Communication rounds

$$O\left(\sqrt{\frac{L}{\lambda}} \log\left(\frac{LR^2}{\varepsilon}\right)\right) \quad \text{Variance reduction}$$

$$O\left(\sqrt{\frac{\delta}{\lambda}} \log\left(\frac{LR^2}{\varepsilon}\right)\right) \quad \|\nabla^2 F_k(x) - \nabla^2 F(x)\| \leq \delta$$

δ -Similarity

Is this bound tight? That is, can we propose such algorithm that works according to this bound? What is a lower bound?

The answer is - this is the lower bound (up to a log-factors), but is this bound tight or not? - until 2022 it was an open question!

Lower bound for distributed convex optimization under similarity

Communication Complexity of Distributed Convex Learning and Optimization

Yossi Arjevani

Weizmann Institute of Science

Rehovot 7610001, Israel

yossi.arjevani@weizmann.ac.il

Ohad Shamir

Weizmann Institute of Science

Rehovot 7610001, Israel

ohad.shamir@weizmann.ac.il

Abstract

We study the fundamental limits to communication-efficient distributed methods for convex learning and optimization, under different assumptions on the information available to individual machines, and the types of functions considered. We identify cases where existing algorithms are already worst-case optimal, as well as cases where room for further improvement is still possible. Among other things, our results indicate that without similarity between the local objective functions (due to statistical data similarity or otherwise) many communication rounds may be required, even if the machines have unbounded computational power.

Lower communication rounds bound for distributed convex optimization under similarity

We consider the problem of distributed convex learning and optimization, where a set of m machines, each with access to a different local convex function $F_i : \mathbb{R}^d \mapsto \mathbb{R}$ and a convex domain $\mathcal{W} \subseteq \mathbb{R}^d$, attempt to solve the optimization problem

$$\min_{\mathbf{w} \in \mathcal{W}} F(\mathbf{w}) \quad \text{where} \quad F(\mathbf{w}) = \frac{1}{m} \sum_{i=1}^m F_i(\mathbf{w}). \quad (1)$$

A prominent application is empirical risk minimization, where the goal is to minimize the average loss over some dataset, where each machine has access to a different subset of the data. Letting $\{\mathbf{z}_1, \dots, \mathbf{z}_N\}$ be the dataset composed of N examples, and assuming the loss function $\ell(\mathbf{w}, \mathbf{z})$ is convex in \mathbf{w} , then the empirical risk minimization problem $\min_{\mathbf{w} \in \mathcal{W}} \frac{1}{N} \sum_{i=1}^N \ell(\mathbf{w}, \mathbf{z}_i)$ can be written as in Eq. (1), where $F_i(\mathbf{w})$ is the average loss over machine i 's examples.

Lower communication rounds bound for distributed convex optimization under similarity

Definition 1. *We say that a set of quadratic functions*

$$F_i(\mathbf{w}) := \mathbf{w}^\top A_i \mathbf{w} + \mathbf{b}_i \mathbf{w} + c_i, \quad A_i \in \mathbb{R}^{d \times d}, \mathbf{b}_i \in \mathbb{R}^d, c_i \in \mathbb{R}$$

are δ -related, if for any $i, j \in \{1 \dots k\}$, it holds that

$$\|A_i - A_j\| \leq \delta, \quad \|\mathbf{b}_i - \mathbf{b}_j\| \leq \delta, \quad |c_i - c_j| \leq \delta$$

Assumption 1. *For each machine j , define a set $W_j \subset \mathbb{R}^d$, initially $W_j = \{\mathbf{0}\}$. Between communication rounds, each machine j iteratively computes and adds to W_j some finite number of points \mathbf{w} , each satisfying*

$$\begin{aligned} & \gamma \mathbf{w} + \nu \nabla F_j(\mathbf{w}) \in \text{span} \left\{ \mathbf{w}', \nabla F_j(\mathbf{w}'), (\nabla^2 F_j(\mathbf{w}') + D) \mathbf{w}'', (\nabla^2 F_j(\mathbf{w}') + D)^{-1} \mathbf{w}'' \mid \right. \\ & \left. \mathbf{w}', \mathbf{w}'' \in W_j, D \text{ diagonal}, \nabla^2 F_j(\mathbf{w}') \text{ exists}, (\nabla^2 F_j(\mathbf{w}') + D)^{-1} \text{ exists} \right\}. \end{aligned} \quad (2)$$

for some $\gamma, \nu \geq 0$ such that $\gamma + \nu > 0$. After every communication round, let $W_j := \cup_{i=1}^m W_i$ for all j . The algorithm's final output (provided by the designated machine j) is a point in the span of W_j .

Lower communication rounds bound for distributed convex optimization under similarity

Definition 1. We say that a set of quadratic functions

$$F_i(\mathbf{w}) := \mathbf{w}^\top A_i \mathbf{w} + \mathbf{b}_i^\top \mathbf{w} + c_i, \quad A_i \in \mathbb{R}^{d \times d}, \mathbf{b}_i \in \mathbb{R}^d, c_i \in \mathbb{R}$$

are δ -related, if for any $i, j \in \{1 \dots k\}$, it holds that

$$\|A_i - A_j\| \leq \delta, \quad \|\mathbf{b}_i - \mathbf{b}_j\| \leq \delta, \quad |c_i - c_j| \leq \delta$$

We begin by presenting a lower bound when the local functions F_i are strongly-convex and smooth:

Theorem 1. For any even number m of machines, any distributed algorithm which satisfies Assumption [1](#), and for any $\lambda \in [0, 1)$, $\delta \in (0, 1)$, there exist m local quadratic functions over \mathbb{R}^d (where d is sufficiently large) which are 1-smooth, λ -strongly convex, and δ -related, such that if $\mathbf{w}^* = \arg \min_{\mathbf{w} \in \mathbb{R}^d} F(\mathbf{w})$, then the number of communication rounds required to obtain $\hat{\mathbf{w}}$ satisfying $F(\hat{\mathbf{w}}) - F(\mathbf{w}^*) \leq \epsilon$ (for any $\epsilon > 0$) is at least

$$\frac{1}{4} \left(\sqrt{1 + \delta \left(\frac{1}{\lambda} - 1 \right)} - 1 \right) \log \left(\frac{\lambda \|\mathbf{w}^*\|^2}{4\epsilon} \right) - \frac{1}{2} = \Omega \left(\sqrt{\frac{\delta}{\lambda}} \log \left(\frac{\lambda \|\mathbf{w}^*\|^2}{\epsilon} \right) \right)$$

if $\lambda > 0$, and at least $\sqrt{\frac{3\delta}{32\epsilon}} \|\mathbf{w}^*\| - 2$ if $\lambda = 0$.

Lower communication rounds bound for distributed convex optimization under similarity (two machines)

$$F_1(\mathbf{w}) = \frac{\delta(1-\lambda)}{4} \mathbf{w}^\top A_1 \mathbf{w} - \frac{\delta(1-\lambda)}{2} \mathbf{e}_1^\top \mathbf{w} + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

$$F_2(\mathbf{w}) = \frac{\delta(1-\lambda)}{4} \mathbf{w}^\top A_2 \mathbf{w} + \frac{\lambda}{2} \|\mathbf{w}\|^2, \quad \text{where}$$

$$A_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & 1 & -1 & 0 & 0 & 0 & \dots \\ 0 & -1 & 1 & 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & 1 & -1 & 0 & \dots \\ 0 & 0 & 0 & -1 & 1 & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}, \quad A_2 = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & \dots \\ -1 & 1 & 0 & 0 & 0 & 0 & \dots \\ 0 & 0 & 1 & -1 & 0 & 0 & \dots \\ 0 & 0 & -1 & 1 & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & 1 & -1 & \dots \\ 0 & 0 & 0 & 0 & -1 & 1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

Upper communication rounds bound for distributed convex optimization under similarity

Lower bound (Arjevani-Shamir, 2015):

$$\Omega \left(\sqrt{\frac{\delta}{\lambda}} \log \left(\frac{\lambda R^2}{\varepsilon} \right) \right)$$

Upper bound (for DISCO, 2019):

<http://proceedings.mlr.press/v37/zhangb15.pdf>

$$\tilde{O} \left(\sqrt{\frac{\delta}{\lambda}} \left(\log \left(\frac{\lambda R^2}{\varepsilon} \right) + \frac{L_2^2 \Delta F}{\lambda^3} \right) \right)$$

Upper bound (for SONATA, 2019):

[arXiv:1905.02637v2](https://arxiv.org/abs/1905.02637v2)

$$\tilde{O} \left(\frac{\delta}{\lambda} \log \left(\frac{\lambda R^2}{\varepsilon} \right) \right)$$

Upper bound (for SPAG, 2020):

<http://proceedings.mlr.press/v119/hendrikx20a/hendrikx20a.pdf>

$$\tilde{O} \left(\frac{\delta}{\lambda} \log \left(\frac{\lambda R^2}{\varepsilon} \right) \right)$$

Upper bound (Agafonov et al., 2021):

[arXiv:2103.14392](https://arxiv.org/abs/2103.14392)

$$\tilde{O} \left(\sqrt{\frac{\delta}{\lambda}} \log \left(\frac{\lambda R^2}{\varepsilon} \right) + \left(\frac{L_2^2 \Delta F}{\lambda^3} \right)^{1/6} \right)$$

Upper communication rounds bound for distributed convex optimization under similarity

Lower bound (Arjevani-Shamir, 2015):

$$\Omega \left(\sqrt{\frac{\delta}{\lambda}} \log \left(\frac{\lambda R^2}{\varepsilon} \right) \right)$$

Upper bound (for Catalyst-SONATA, 2022):

Tian, Y., Scutari, G., Cao, T., & Gasnikov, A. (2022, May). Acceleration in distributed optimization under similarity. In International Conference on Artificial Intelligence and Statistics (pp. 5721–5756). PMLR.

$$\tilde{O} \left(\sqrt{\frac{\delta}{\lambda}} \log \left(\frac{\lambda R^2}{\varepsilon} \right) \right)$$

Upper bound (Optimal Sliding, 2022):

Kovalev, D., Beznosikov, A., Borodich, E., Gasnikov, A., & Scutari, G. (2022). Optimal Gradient Sliding and its Application to Distributed Optimization Under Similarity. arXiv preprint arXiv:2205.15136.

$$O \left(\sqrt{\frac{\delta}{\lambda}} \log \left(\frac{\lambda R^2}{\varepsilon} \right) \right)$$

Scheme of the prove

Upper bound (Optimal Sliding, 2022):

Kovalev, D., Beznosikov, A., Borodich, E., Gasnikov, A., & Scutari, G. (2022). Optimal Gradient Sliding and its Application to Distributed Optimization Under Similarity. arXiv preprint arXiv:2205.15136.

$$O\left(\sqrt{\frac{\delta}{\lambda}} \log\left(\frac{\lambda R^2}{\varepsilon}\right)\right)$$

$$\min_{x \in Q} \left[\bar{f}(x) := \frac{1}{m} \sum_{k=1}^m \bar{f}_k(x) := \frac{1}{m} \sum_{k=1}^m \frac{1}{s} \sum_{j=1}^s f(x, \xi^{k,j}) \right].$$

To use similarity we describe **Accelerated gradient sliding** for unconstrained composite optimization problem:

$$\min_{x \in \mathbb{R}^n} [\bar{f}(x) := g(x) + h(x)],$$

where $g(x)$ has L_g -Lipschitz continuous gradient, $h(x)$ is convex and has L_h -Lipschitz continuous gradient ($L_g \leq L_h$); $\bar{f}(x)$ is μ -strongly convex function in 2-norm. Note that we do not assume $g(x)$ to be convex!

Optimal Sliding Algorithm

$$\min_{x \in \mathbb{R}^n} [\bar{f}(x) := g(x) + h(x)]$$

$$\tilde{x}^t = \tau x^t + (1 - \tau)x_f^t,$$

$$x_f^{t+1} \approx \operatorname{argmin}_{x \in \mathbb{R}^n} [A^t(x) := g(\tilde{x}^t) + \langle \nabla g(\tilde{x}^t), x - \tilde{x}^t \rangle + L_g \|x - \tilde{x}^t\|_2^2 + h(x)],$$

which means

$$\|\nabla A^t(x_f^{t+1})\|_2^2 \leq \frac{L_g^2}{3} \left\| \tilde{x}^t - \arg \min_{x \in \mathbb{R}^n} A^t(x) \right\|_2^2,$$

$$x^{t+1} = x^t + \eta \mu (x_f^{t+1} - x^t) - \eta \nabla \bar{f}(x_f^{t+1}),$$

where

$$\tau = \min \left\{ 1, \frac{\sqrt{\mu}}{2\sqrt{L_g}} \right\}, \quad \eta = \min \left\{ \frac{1}{2\mu}, \frac{1}{2\sqrt{\mu L_g}} \right\}.$$

Properties of the Algorithm

$$\min_{x \in \mathbb{R}^n} [\bar{f}(x) := g(x) + h(x)]$$

This algorithm (with output point x^N) has an iteration complexity

$$\tilde{O}\left(\sqrt{\frac{L_g}{\mu}}\right)$$

and solves several tasks at once:

- **(simple acceleration)** If $h(x) \equiv 0$ this algorithm becomes an ordinary accelerated method with

$$x_f^{t+1} = \tilde{x}^t - \frac{1}{2L_p} \nabla g(\tilde{x}^t);$$

- **(Catalyst)** If $g(x) \equiv 0$ this algorithm becomes a Catalyst-type proximal envelop [72], but less sensitive to the accuracy of the solution of (1.59)

$$x_f^{t+1} \approx \operatorname{argmin}_{x \in \mathbb{R}^n} [A^t(x) := g(\tilde{x}^t) + \langle \nabla g(\tilde{x}^t), x - \tilde{x}^t \rangle + L_g \|x - \tilde{x}^t\|_2^2 + h(x)], \quad (1.59)$$

Properties of the Algorithm

$$\min_{x \in \mathbb{R}^n} [\bar{f}(x) := g(x) + h(x)]$$

$$x_f^{t+1} \approx \operatorname{argmin}_{x \in \mathbb{R}^n} [A^t(x) := g(\tilde{x}^t) + \langle \nabla g(\tilde{x}^t), x - \tilde{x}^t \rangle + L_g \|x - \tilde{x}^t\|_2^2 + h(x)], \quad (1.59)$$

- **(Sliding)** If we apply to (1.59) Accelerated gradient sliding with $g(x) := h(x)$ then obtain the total complexity of $\nabla h(x)$ oracle as

$$\tilde{O}\left(\sqrt{\frac{L_g}{\mu}}\right) \cdot O\left(\sqrt{\frac{L_g + L_h}{L_g}}\right) = \tilde{O}\left(\sqrt{\frac{L_h}{\mu}}\right).$$

That is, we have split the complexity of considered composite problem to the complexities correspond to the separate problems:

$$\tilde{O}\left(\sqrt{\frac{L_g}{\mu}}\right) \quad \text{for } \#\nabla g(x) \quad \text{and} \quad \tilde{O}\left(\sqrt{\frac{L_h}{\mu}}\right) \quad \text{for } \#\nabla h(x).$$

Main idea

$$\min_{x \in Q} \left[\bar{f}(x) := \frac{1}{m} \sum_{k=1}^m \bar{f}_k(x) := \frac{1}{m} \sum_{k=1}^m \frac{1}{s} \sum_{j=1}^s f(x, \xi^{k,j}) \right].$$

Let us rewrite the empirical problem as follows

$$\min_{x \in \mathbb{R}^n} [\bar{f}(x) := (\bar{f}(x) - \bar{f}_1(x)) + \bar{f}_1(x)].$$

That is, we have split the complexity of considered composite problem to the complexities correspond to the separate problems:

$$\tilde{O}\left(\sqrt{\frac{L_g}{\mu}}\right) \quad \text{for } \#\nabla g(x) \quad \text{and} \quad \tilde{O}\left(\sqrt{\frac{L_h}{\mu}}\right) \quad \text{for } \#\nabla h(x).$$

Denoting the first sum as $g(x)$ and the second one as $h(x)$ we can use Sliding trick to split the complexities. Note that we significantly use the fact that in this scheme $g(x)$ is not necessarily convex! So it remains only to notice that described Accelerated gradient sliding under this choice of $g(x)$ and $h(x)$ has a natural distributed interpretation. It gives at the end a distributed algorithm that works according to the lower bounds for communications and oracle calls per node complexities under similarity [7]. Due to the statistical (i.i.d.) nature of $\{\xi^{k,j}\}$ (statistical similarity) one may expect that [51]: $L_g \propto s^{-1/2}$.

Distributed interpretation

$$\min_{x \in Q} \left[\bar{f}(x) := \frac{1}{m} \sum_{k=1}^m \bar{f}_k(x) := \frac{1}{m} \sum_{k=1}^m \frac{1}{s} \sum_{j=1}^s f(x, \xi^{k,j}) \right].$$

Let us rewrite the empirical problem as follows

$$\min_{x \in \mathbb{R}^n} [\bar{f}(x) := (\bar{f}(x) - \bar{f}_1(x)) + \bar{f}_1(x)].$$

That is, we have split the complexity of considered composite problem to the complexities correspond to the separate problems:

$$\tilde{O}\left(\sqrt{\frac{L_g}{\mu}}\right) \quad \text{for } \#\nabla g(x) \quad \text{and} \quad \tilde{O}\left(\sqrt{\frac{L_h}{\mu}}\right) \quad \text{for } \#\nabla h(x).$$

Indeed, we can assign the node number 1 to be a master node that minimize at each iteration (1.59) with $g(x) := \bar{f}(x) - \bar{f}_1(x)$ and $h(x) := \bar{f}_1(x)$. It is obvious, that $h(x)$ is available to the master node and $\nabla g(\tilde{x}^t)$ can be available due to communications of the master node with the other ones. At each round of communications k -th node sends $\nabla \bar{f}_k(\tilde{x}^t)$ to the master node and receive in return x_f^{t+1} , which is calculated at the master node.

$$x_f^{t+1} \approx \operatorname{argmin}_{x \in \mathbb{R}^n} [A^t(x) := g(\tilde{x}^t) + \langle \nabla g(\tilde{x}^t), x - \tilde{x}^t \rangle + L_g \|x - \tilde{x}^t\|_2^2 + h(x)], \quad (1.59)$$

Convex-Concave Saddle-Point Problems

Variance reduction

Optimal Algorithms for Decentralized Stochastic Variational Inequalities

Dmitry Kovalev¹ Aleksandr Beznosikov² Abdurakhmon Sadiev² Michael Persiianov² Peter Richtárik¹
Alexander Gasnikov²

Abstract

Variational inequalities are a formalism that includes games, minimization, saddle point, and equilibrium problems as special cases. Methods for variational inequalities are therefore universal approaches for many applied tasks, including machine learning problems. This work concentrates on the decentralized setting, which is increasingly important but not well understood. In particular, we consider decentralized stochastic (sum-type) variational inequalities over fixed and time-varying networks. We present lower complexity bounds for both communication and local iterations and construct optimal algorithms that match these lower bounds. Our algorithms are the best among the available literature not only in the decentralized stochastic case, but also in the decentralized deterministic and non-distributed stochastic cases. Experimental results confirm the effectiveness of the presented algorithms.

1.1. Applications of variational inequalities

In recent years, there has been a significant increase of research activity in the study of variational inequalities due to new connections to reinforcement learning (Omidshafiei et al., 2017; Jin & Sidford, 2020), adversarial training (Madry et al., 2018), and GANs (Goodfellow et al., 2014). In particular, Daskalakis et al. (2017); Gidel et al. (2018); Mertikopoulos et al. (2018); Chavdarova et al. (2019); Liang & Stokes (2019); Peng et al. (2020) show that even if one considers the classical (in the variational inequalities literature) regime involving monotone and strongly monotone inequalities, it is possible to obtain insights, methods and recommendations useful for the GAN community.

In addition to the above modern applications, and besides their many classical applications in applied mathematics that include economics, equilibrium theory, game theory and optimal control (Facchinei & Pang, 2007), variational inequalities remain popular in supervised learning (with non-separable loss (Joachims, 2005); with non-separable regularizer (Bach et al., 2011)), unsupervised learning (discriminative clustering (Xu et al., 2005); matrix factorization

Convex-Concave Saddle-Point Problems

Similarity

Optimal Gradient Sliding and its Application to Distributed Optimization Under Similarity

Dmitry Kovalev
KAUST

dakovalev1@gmail.com

Aleksandr Beznosikov
MIPT

anbeznosikov@gmail.com

Ekaterina Borodich
MIPT

borodich.ed@phystech.edu

Alexander Gasnikov
MIPT

gasnikov@yandex.ru

Gesualdo Scutari

Purdue University

gscutari@purdue.edu

Abstract

We study structured convex optimization problems, with additive objective $r := p + q$, where r is $(\mu$ -strongly) convex, q is L_q -smooth and convex, and p is L_p -smooth, possibly nonconvex. For such a class of problems, we proposed an inexact accelerated gradient sliding method that can skip the gradient computation for one of these components while still achieving optimal complexity of gradient calls of p and q , that is, $\mathcal{O}(\sqrt{L_p/\mu})$ and $\mathcal{O}(\sqrt{L_q/\mu})$, respectively. This result is much sharper than the classic black-box complexity $\mathcal{O}(\sqrt{(L_p + L_q)/\mu})$, especially when the difference between L_q and L_p is large. We then apply the proposed method to solve distributed optimization problems over master-worker architectures, under agents' function similarity, due to statistical data similarity or otherwise. The distributed algorithm achieves for the first time lower complexity bounds on *both* communication and local gradient calls, with the former having being a long-standing open problem. Finally the method is extended to distributed saddle-problems (under function similarity) by means of solving a class of variational inequalities, achieving lower communication and computation complexity bounds.