Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

On the line-search gradient methods for stochastic optimization ⋆

Darina Dvinskikh [**] **Aleksandr Ogaltsov** [*]
Alexander Gasnikov [****] **Pavel Dvurechensky** [***]
Vladimir Spokoiny [†]

[*] *Higher school of economics, Moscow*
Antiplagiat Company (e-mail: aogalcov@hse.ru)
[**] *Weierstrass Institute for Applied Analysis and Stochastics, Berlin,*
Moscow Institute of Physics and Technology, Dolgoprudny, Russia,
Institute for Information Transmission Problems RAS, Moscow
(e-mail: darina.dvinskikh@wias-berlin.de)
[***] *Weierstrass Institute for Applied Analysis and Stochastics, Berlin,*
Institute for Information Transmission Problems RAS, Moscow
(e-mail: pavel.dvurechensky@wias-berlin.de)
[****] *Moscow Institute of Physics and Technology, Dolgoprudny, Russia,*
Institute for Information Transmission Problems RAS, Moscow,
Higher school of economics (e-mail: gasnikov@yandex.ru)
[†] *Weierstrass Institute for Applied Analysis and Stochastics, Berlin,*
Higher school of economics(e-mail: spokoiny@wias-berlin.de)

Abstract: We consider several line-search based gradient methods for stochastic optimization: a gradient and accelerated gradient methods for convex optimization and gradient method for non-convex optimization. The methods simultaneously adapt to the unknown Lipschitz constant of the gradient and variance of the stochastic approximation for the gradient. The focus of this paper is to numerically compare such methods with state-of-the-art adaptive methods which are based on a different idea of taking norm of the stochastic gradient to define the stepsize, e.g., AdaGrad and Adam.

Keywords: convex and non-convex optimization, stochastic optimization, first-order method, adaptive method, gradient descent, complexity bounds, mini-batch.

1. INTRODUCTION

In this paper we consider unconstrained minimization problem

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1}$$

where $f(x)$ is a smooth, possibly non-convex function with L-Lipschitz continuous gradient. We say that a function $f : E \to \mathbb{R}$ has a L-Lipschitz continuous gradient if it is continuously differentiable and its gradient satisfies

$$f(y) \le f(x) + \langle \nabla f(x), y - x\rangle + \frac{L}{2}\|x - y\|_2^2, \quad \forall\, x, y \in E.$$

We assume that the access to the objective f is given through stochastic oracle $\nabla f(x, \xi)$, where ξ is a random variable. The main assumptions on the stochastic oracle are standard for stochastic approximation literature

$$\mathbb{E}\nabla f(x,\xi) = \nabla f(x), \;\; \mathbb{E}\left(\|\nabla f(x,\xi) - \nabla f(x)\|_2^2\right) \le D.$$

In papers Duchi et al. (2011); Byrd et al. (2012); Friedlander and Schmidt (2012); Kingma and Ba (2015); Iusem et al. (2019); Gasnikov (2017); Levy et al. (2018); Deng et al. (2018); Ogaltsov and Tyurin (2019); Ward et al. (2019); Bach and Levy (2019); Kamzolov et al. (2020); Dvurechensky et al. (2020) there proposed different approaches to choose in an adaptive manner L and D, see Table for details.

In this paper we extend the idea of Armijo-type line search in variants from Gasnikov (2017); Ogaltsov and Tyurin (2019) for the adaptive methods for convex and non-convex stochastic optimization. Surprisingly, the adaptation is needed not to each parameter separately, but to the ratio D/L, which can be considered as signal to noise ratio or an effective Lipschitz constant of the gradient in this case. We propose an accelerated and non-accelerated gradient descent for stochastic convex optimization and a gradient method for stochastic non-convex optimization. Our methods are flexible enough to use optimal choice of mini-batch size without additional information on the problem. Moreover, our procedure allows an increase of the stepsize, which accelerated the methods in the areas where the Lipschitz constant is small. Also, as opposed to the existing methods, our algorithms do not need to know neither the distance to the solution, nor a set of complicated hyperparameters, which are usually fine-tuned

⋆ The research of A. Gasnikov and P. Dvurechensky was partially supported by Russian Foundation for Basic Research project 18-31-20005 mol-a-ved. The research of D. Dvinskikh was supported by the Ministry of Science and Higher Education of the Russian Federation (Goszadaniye) no 075-00337-20-03.

Paper	N-C [1]	N-Ac	Ac	Prf	Btch	Par
Duchi et al.'11	×	√	×	√	×	×
Byrd et al.'12	×	√	×	×	×	×
Friedlander et al.'12	×	√	×	×	×	×
Kingma & Ba'15	×	√	√	×	×	×
Iusem et al.'19	×	√	×	√	×	√
Gasnikov'17	×	√	×	×	√	√
Levy et al.'18	×	√	×	√	×	×
Deng et al.'18	×	×	√	×	×	×
Ogaltsov et al.'19	×	×	√	×	√	√
Ward et al.'19	√	×	×	√	×	√
Bach & Levy'19	×	√	×	√	×	×
This paper	√	√	√	×	√	√

by multiple repetition of minimization process. Moreover, since our methods are based on inexact oracle model (see e.g. Devolder et al. (2014); Gasnikov and Dvurechensky (2016); Dvurechensky and Gasnikov (2016)), they are adaptive not only for a stochastic error, but also for deterministic, e.g. non-smoothness of the problem. This means that our methods are universal for smooth and non-smooth optimization Nesterov (2015); Yurtsever et al. (2015); Dvurechensky (2017). Our focus in this paper is to demonstrate in the experiments that our methods work faster than state-of-the-art methods Duchi et al. (2011); Kingma and Ba (2015). The rigorous proofs are deferred to a separate paper.

The paper is structured as follows. In Sect. 2 we present adaptive stochastic algorithms based on stochastic gradient method to solve a problem of type (1) with convex objective function. Sect. 3 renews Sect. 2 for non-convex objective function. Finally, in Sect. 4 we show numerical experiments supporting the theory in above sections.

2. STOCHASTIC CONVEX OPTIMIZATION

In this section we solve problem (1) for convex objective by adaptive algorithm which does not need the information about Lipschitz constant. Then we comment on its acceleration and practical implementation.

2.1 Adaptive algorithm

We assume that the constant L may be unknown. If the true variance D is unavailable we use its upper bound $D_0 \geq D$. We provide adaptive algorithm (Alg. 1) which iteratively tunes the Lipschitz constant. Importantly, the approximation of the Lipschitz constant used by the algorithm may decrease as iteration go, leading to larger steps and faster convergence. Further, we comment on its rates of convergence.

From Lipschitz continuity of $\nabla f(x)$ we have

$$f(x^{k+1}) \leq f(x^k) + \langle \nabla f(x^k), x^{k+1} - x^k \rangle + \frac{L_{k+1}}{2}\|x^{k+1} - x^k\|_2^2. \quad (3)$$

[1] N-C stands for availability of an algorithm for non-convex optimization, N-Ac for a non-accelerated algorithm for convex optimization, Ac for accelerated algorithm for convex optimization, Prf for proof of the convergence rate, Btch for possibility to adaptively choose batch size without knowing other parameters, Par for non-necessity to know or tune hyperparameters like distance to the solution for choosing the stepsize.

Algorithm 1 Adaptive Stochastic Gradient Descent

Require: Number of iterations N, accuracy ε, D_0, initial guess L_0.

for $k = 0, \ldots, N-1$ **do**
  $L_{k+1} := L_k/4$
  repeat
    $L_{k+1} := 2L_{k+1}$
    Calculate batch size $r_{k+1} = \max\left\{\frac{D_0}{L_{k+1}\varepsilon},\ 1\right\}$

$$x^{k+1} = x^k - \frac{1}{2L_{k+1}}\nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}) \quad (2)$$

  until

$$f(x^{k+1}) \leq f(x^k) + \langle \nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}), x^{k+1} - x^k \rangle + L_{k+1}\|x^{k+1} - x^k\|_2^2 + \varepsilon/2$$

end for

Ensure: $\bar{x}_N = \frac{1}{N}\sum_{k=1}^N x^k$.

By Cauchy–Schwarz inequality and since $ab \leq \frac{a^2}{2} + \frac{b^2}{2}$ for any a, b, we get

$$\langle \nabla f(x^k) - \nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}), x^{k+1} - x^k \rangle \leq \frac{1}{2L_{k+1}}\|\nabla f(x^k) - \nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}})\|_2^2 + \frac{L_{k+1}}{2}\|x^{k+1} - x^k\|_2^2. \quad (4)$$

Then we add and subtract $\nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}})$ in (3). Using (4) and (2) we get

$$f(x^{k+1}) - f(x^k) \leq -\frac{1}{4L_{k+1}^2}\|\nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}})\|_2^2 + \frac{1}{2L_{k+1}}\|\nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}) - \nabla f(x^k)\|_2^2. \quad (5)$$

From (2) we have for any x

$$\|x^{k+1} - x\|_2^2 = \|x^k - x - \frac{1}{2L_{k+1}}\nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}})\|_2^2 = \|x^k - x\|_2^2 + \frac{1}{4L_{k+1}^2}\|\nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}})\|_2^2 - \frac{1}{L_{k+1}}\langle \nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}), x^k - x \rangle. \quad (6)$$

From (5) and (6) we get

$$\langle \nabla^{r_{k+1}} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}), x^k - x \rangle \leq f(x^k) - f(x^{k+1}) + L_{k+1}\|x^k - x\|_2^2 - L_{k+1}\|x^{k+1} - x\|_2^2 + \varepsilon/2, \quad (7)$$

where we used batch size $r_{k+1} = \max\left\{\frac{D_0}{L_{k+1}\varepsilon},\ 1\right\}$ to fulfill $\mathbb{E}\|\nabla^{r+1} f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r}) - \nabla f(x^k)\|_2^2 = \varepsilon L_{k+1}$.

Since L_{k+1} is random, r_{k+1} will be random as well and, consequently, the total number of oracle calls T is not precisely determined. Let us choose it according to the number of oracle calls for non-adaptive counterpart of Algorithm 1

$$T = \sum_{k=1}^{N-1} r_{k+1} = O\left(\frac{D_0 R^2}{\varepsilon^2}\right). \quad (8)$$

This number of oracle calls (8) can be provided by choosing the last batch size r_N as a residual of the sum (8) and calculate the last Lipschitz constant $L_N = \frac{D_0}{r_N \varepsilon}$. In practice, we do not need to limit ourselves by fixing the number of oracle calls T.

From the convexity of f we have

$$f(x^k) - f(x) \leq \langle \nabla f(x), x^k - x \rangle. \quad (9)$$

Then from (9) we get

$$\langle \nabla^{r_{k+1}} f(x, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}), x^k - x \rangle \geq f(x^k) - f(x) + \langle \nabla^{r_{k+1}} f(x, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}) - \nabla f(x^k),\ x^k - x \rangle.$$

From this and (7) we have

$$\frac{1}{L_{k+1}}(f(x^k) - f(x)) + \frac{1}{L_{k+1}} \langle \nabla^{r_{k+1}} f(x, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}) - \nabla f(x^k), x^k - x \rangle \leq \|x^k - x\|_2^2 - \|x^{k+1} - x\|_2^2 + \frac{1}{L_{k+1}} \left(f(x^k) - f(x^{k+1}) \right) + \frac{\varepsilon}{2L_{k+1}}.$$

We notice that the following sum $\sum_{k=0}^{N-1} \frac{1}{L_{k+1}} \langle \nabla^{r_{k+1}} f(x^k, \{\xi_l^{r_{k+1}}\}_{l=1}^{r_{k+1}}) - \nabla f(x^k), x^k - x \rangle$ is not the sum of martingale-differences and therefore the total expectation is not zero, since r_k is random. Thus, we cannot guarantee that Alg. 1 converges in $O\left(\frac{LR^2}{\varepsilon}\right)$ iterations. However, numerical experiments are in a good agreement with the provided complexity bound.

2.2 Accelerated adaptive algorithm

To compare our complexity bounds for adaptive stochastic gradient descent with the bounds for accelerated variant of our algorithm we refer to Ogaltsov and Tyurin (2019). For the reader convenience we provide that accelerated algorithm in a simpler form and complexity bounds presented there without proof.

Algorithm 2 Adaptive Stochastic Accelerated Gradient Method

Require: Number of iterations N, D_0 accuracy ε, $\Omega \geq 1$, $A_0 = 0$, initial guess L_0.

for $k = 0, \ldots, N-1$ **do**
  $L_{k+1} := L_k/4$
  repeat
    $L_{k+1} := 2L_{k+1}$
    $\alpha_{k+1} = \frac{1+\sqrt{1+8A_kL_{k+1}}}{4L_{k+1}}$; $A_{k+1} = A_k + \alpha_{k+1}$
    Calculate batch size $r_{k+1} = \max\left\{\frac{\Omega\alpha_{k+1}D_0}{\varepsilon},\ 1\right\}$
    $y^{k+1} = (\alpha_{k+1}u^k + A_kx^k)/A_{k+1}$
    $u^{k+1} = u^k - \alpha_{k+1}\nabla^{r_{k+1}} f(y^{k+1}, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}})$
    $x^{k+1} = (\alpha_{k+1}u^{k+1} + A_kx^k)/A_{k+1}$
  until

$$f(x^{k+1}) \leq f(y^{k+1}) + \langle \nabla^{r_{k+1}} f(y^{k+1}, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}), x^{k+1} - y^{k+1} \rangle + L_{k+1}\|x^{k+1} - y^{k+1}\|_2^2 + \Omega D_0/(L_{k+1}r_{k+1}) \quad (10)$$

end for
Ensure: x^N

For Algorithm 2 the number of oracle calls T will be the same as for the non-accelerated version of the algorithm while the number of iterations will be smaller: $N = O\left(\sqrt{LR^2/\varepsilon}\right)$. Both these bounds are optimal Woodworth et al. (2018).

Unfortunately, to prove these bounds we also met the problem of martingale-differences mentioned above. We expect that original technique from the paper Iusem et al. (2019) sheds light on how one can try to resolve it and we defer the complete proof to a future version of this paper.

2.3 Practical implementation of adaptive algorithms

Next we comment on applicability of Alg. 1 and Alg. 2 in real problems. Generally, in case when the exact gradients of function $f(x^k)$ is unavailable, function values itself of $f(x^k)$ are also unavailable. It holds, e.g., in stochastic optimization problem, where the objective is presented by its expectation

$$f(x) = \mathbb{E} f(x, \xi). \quad (11)$$

In this case we estimate the function as a sample average $f(x, \{\xi_l\}_{l=1}^r) = \frac{1}{r}\sum_{l=1}^r f(x, \xi_l)$ and use it in adaptive procedures. In this case we interpret L_k as the worst constant among all Lipschitz constants for $f(x, \xi)$ with different realization of ξ. Indeed, if L_{k+1} satisfies the following

$$f(x^{k+1}, \xi^{k+1}) \leq f(x^k, \xi^{k+1}) + \langle \nabla f(x^k, \xi), x^{k+1} - x^k \rangle + L_{k+1}\|x^{k+1} - x^k\|_2^2 + \varepsilon/2.$$

Then it satisfies

$$f(x^{k+1}, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}) \leq f(x^k, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}) + \langle \nabla^{r_{k+1}} f(x, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}}), x^{k+1} - x^k \rangle + L_{k+1}\|x^{k+1} - x^k\|_2^2 + \varepsilon/2. \quad (12)$$

If, e.g, (11) holds we replace adaptive procedure in the algorithms by (12).

We also comment on batch size. If the batch size r_k decreases during the process of L_k selection, we preserve r_k from the previous iteration in order not to recalculate stochastic approximation $\nabla^{r_{k+1}} f(x, \{\xi_l^{k+1}\}_{l=1}^{r_{k+1}})$.

All these remarks remain true also in non-convex case.

3. STOCHASTIC NON-CONVEX OPTIMIZATION

In this section we assume that the objective f may be non-convex. We consider adaptive algorithm and provide the complexity bounds for it.

Algorithm 3 Adaptive Non-convex Stochastic Gradient Descent

Require: Number of iterations N, D_0, accuracy ε, initial guess L_0

Calculate batch size $r = \max\left\{\frac{8D_0}{\varepsilon^2},\ 1\right\}$
for $k = 0, \ldots, N-1$ **do**
  $L_{k+1} := L_k/4$.
  repeat
    $L_{k+1} := 2L_{k+1}$

$$x^{k+1} = x^k - \frac{1}{2L_{k+1}}\nabla^r f(x^k, \{\xi_l^{k+1}\}_{l=1}^r) \quad (13)$$

  until

$$f(x^{k+1}) \leq f(x^k) + \langle \nabla^r f(x^k, \{\xi_l^{k+1}\}_{l=1}^r), x^{k+1} - x^k \rangle + L_{k+1}\|x^{k+1} - x^k\|_2^2 + \frac{\varepsilon^2}{32L_{k+1}} \quad (14)$$

end for
Ensure: $\hat{x} = \arg\min_{k=1,..N} \|\nabla f(x^k)\|_2$.

Theorem 1. Algorithm 3 with expected number of stochastic gradient oracle calls $\tilde{T} = O\left(\frac{D_0 L(f(x^0) - f(x^N))}{\varepsilon^4}\right)$ and

expected number of iterations $\tilde{N} = O\left(\frac{L(f(x^0)-f(x^N))}{\varepsilon^2}\right)$ outputs a point $\hat{x}^{\tilde{N}}$ satisfying

$$\mathbb{E}\|\nabla f(\hat{x}^{\tilde{N}})\|_2^2 \leq \varepsilon^2.$$

Sketch of the Proof. From (14) using (13) we get

$$f(x^{k+1}) - f(x^k) \leq -\frac{1}{4L_{k+1}}\|\nabla^r f(x^k, \{\xi_l^{k+1}\}_{l=1}^r)\|_2^2 + \frac{\varepsilon^2}{32L_{k+1}}. \quad (15)$$

Due to $\|a\|^2 \leq 2\|b\|^2 + 2\|a-b\|^2$ for any $a, b \in \mathbb{R}^n$ we get

$$\|\nabla^r f(x^k, \{\xi_l^{k+1}\}_{l=1}^r)\|_2^2 \geq \frac{1}{2}\|\nabla f(x^k)\|_2^2 - \|\nabla f(x^k) - \nabla^r f(x^k, \{\xi_l^{k+1}\}_{l=1}^r)\|_2^2. \quad (16)$$

From (15) and (16) we have

$$f(x^{k+1}) - f(x^k) \leq -\frac{1}{8L_{k+1}}\|\nabla f(x^k)\|_2^2 + \frac{\varepsilon^2}{32L_{k+1}} + \frac{1}{4L_{k+1}}\|\nabla f(x^k) - \nabla^r f(x^k, \{\xi_l^{k+1}\}_{l=1}^r)\|_2^2.$$

If $\|\nabla f(x^k)\|_2^2 \geq \varepsilon^2$. Then

$$f(x^{k+1}) - f(x^k) \leq (8\delta_{k+1}^2 - 3\varepsilon^2)/(32L_{k+1}). \quad (17)$$

Based on iterated procedure (10) we may expect that $L_{k+1} \leq 2L$. The exact proof of this fact in probability of large deviations terminology was provided in Ogaltsov and Tyurin (2019) (numerical coefficient needs to be corrected). In our work, we limit ourselves by assuming this inequality holds 'in average'.
If $3\varepsilon^2 - 8\delta_{k+1}^2 \geq 0$ we may replace L_{k+1} by $2L$. Therefore, we rewrite (17) with minor changing and after taking the expectation we get

$$\mathbb{E}f(x^{k+1}) - \mathbb{E}f(x^k) \leq (8\mathbb{E}\delta_{k+1}^2 - 2\varepsilon^2)/(64L).$$

Ensuring $\mathbb{E}\delta_{k+1}^2 \geq \frac{\varepsilon^2}{8}$ we obtain

$$\mathbb{E}f(x^{k+1}) - \mathbb{E}f(x^k) \leq -\varepsilon^2/(64L).$$

Summing this over expected number of iteration we get

$$\tilde{N} = 64L(f(x^0) - f(x^*))/\varepsilon^2. \quad (18)$$

This $\tilde{N}$ ensures that for some k we get $\|\nabla f(x^k)\|_2^2 \leq \varepsilon^2$. We choose the batch size according to

$$\mathbb{E}\delta_{k+1}^2 = \mathbb{E}\|\nabla^r f(x^k, \{\xi_l^{k+1}\}_{l=1}^r) - \nabla f(x^k)\|_2^2 = \frac{\varepsilon^2}{8} \leq \frac{D_0}{r}.$$

Consequently, $r = \frac{8D_0}{\varepsilon^2}$. Using the expected number of iterations (18) we get expected number of oracle calls

$$\tilde{T} = \tilde{N}r = 512D_0L(f(x^0) - f(x^N))/\varepsilon^4.$$

□

More accurate proof of Theorem 1 can be done using large deviations technique and sub-Gaussian variance, see Dvurechensky et al. (2018a).

4. EXPERIMENTS

We perform experiments using proposed methods with and without acceleration on convex and non-convex problems and compare results with commonly used methods — Adam, Kingma and Ba (2015) and Adagrad, Duchi et al. (2011). Experiments consist of four problems:

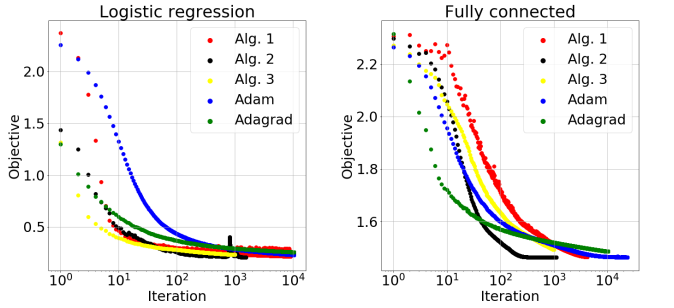
(1) Training logistic regression on MNIST dataset Lecun et al. (1998) (convex problem). Number of optimized parameters is 7850.
(2) Training fully-connected sigmoid-activated neural network with two hidden layers of size 1000 on MNIST dataset (non-convex problem). Number of optimized parameters is 795010.
(3) Training fully-connected *relu-activated* neural network with two hidden layers of size 1000 also on MNIST dataset (*non-differentiable* and non-convex problem). Number of optimized parameters is 795010.
(4) Training small convolutional neural network with three filters and three fully-connected layers on CIFAR10 dataset Krizhevsky (2009) (non-convex problem). Number of optimized parameters is 62000.

Objective for all the problems is cross-entropy function between predicted class distribution and ground-truth class labels. Hyper-parameters set for proposed methods varies depending only on convexity of the problem. Hyper-parameters for all our algorithms in convex case were $D_0 = 0.01, \varepsilon = 10^{-5}, L_k = 100$, and $D_0 = 0.1, L_0 = 1, \varepsilon = 0.002$ for all non-convex problems. This hyperparameter set is chosen experimentally to obtain universal hyperparameters for broad range of settings. Adam and Adagrad had batch size equal to 128, learning rate = 0.001 and $\beta_1 = 0.9, \beta_2 = 0.999$ — these parameters are frequently used in various machine learning tasks and are used in Kingma and Ba (2015). Training set is 60K samples for MNIST dataset and 50K samples for CIFAR10 dataset. Dynamics of objective function value is depicted on Fig 1. Since batch size is variable we also compare algorithms by epochs (one epoch is one full pass through dataset). We also performed grid search of hyper-parameters for all algorithms to compare best versions of the laters. Result for tuned algorithms by epochs for logitic regression and fully connected neural network are in Fig 2. Although Adam performs better than our methods in case of fully connected network, we show that our algorithms are more robust to hyper-parameter choice. Final experiment series is follows. We pick logistic regression for computational simplicity, choice hyper-parameter set and do 10 epochs of optimization procedure than average results for each epoch. Hyper-parameter set for Adam and Adagrad is all pairs of batch size (8, 32, 64, 128, 256, 512, 1024) and learning rate (10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1). Hyper-parameter set for our methods is all combinations of D_0 (10^{-5}, 10^{-4}, 10^{-3}), L_0 (10^4, 10^3, 10^2), ε (10^{-6}, 10^{-5}, 10^{-4}). Median for each epoch (to omit sensibility to outliers) with one standart deviation are on Fig. 3. One can see that underestimate of hyper-parameters for our algorithms does not lead to substantial deviations and that all proposed algorithms outperform Adam and Adagrad in terms of robustness. The code for all algorithms is available, visit. [2]

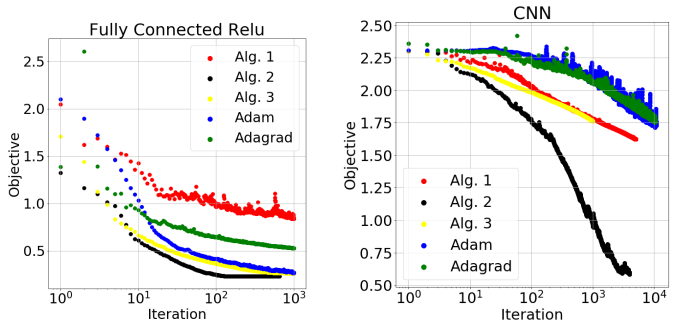
5. CONCLUSION

In this paper we focus on adaptive methods for stochastic convex and non-convex optimization. It would be interesting to combine these ideas with the notion of inexact model

[2] https://github.com/alexo256/Adaptive-Gradient-Descent-for-Convex-and-Non-Convex-Stochastic-Optimization

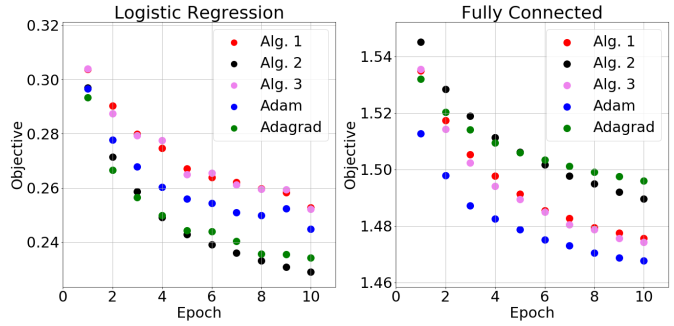


(a) LR Objective (b) FC Objective



(c) FC Relu Objective (d) CNN Objective

Fig. 1. Experiments by iteration



(a) LR Objective (b) FC Objective

Fig. 2. Experiments with tuned algorithms by epoch

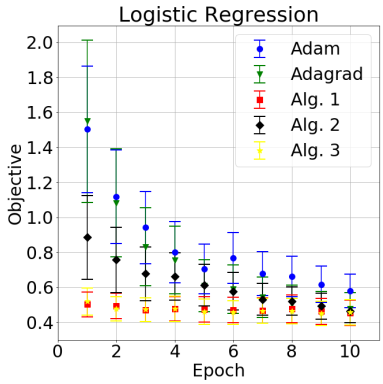


Fig. 3. Robustness to hyperparameters

of the objective function and inexact model Stonyakin et al. (2019b) of the operator in variational inequalities Dvurechensky et al. (2018c); Gasnikov et al. (2019); Stonyakin et al. (2019a) to obtain adaptive and universal methods using stochastic inexact model. We leave this for future work.

It seems that the results of this paper can be generalized on gradient-free method for stochastic optimization Dvurechensky et al. (2017). In particular, for sum-type problems. It would be interesting to apply these methods for stochastic optimization in the context of Wasserstein barycenters Dvurechensky et al. (2018b); Dvinskikh et al. (2019); Kroshnin et al. (2019); Uribe et al. (2018).

Note also, that if we replace step 2 in Alg. 1 and Alg. 2 by $L_{k+1} := L_k/2$, take $r_{k+1} \equiv \max\{2D_0/(L_k\varepsilon), 1\}$ and forbid L_{k+1} to be outside the range $[L_d, L_u]$, where $L_d \equiv L_0 \equiv L_u \bmod 2$, $L_0 \in [L_d, L_u]$, then based on union bound inequality and theory of empirical process Giné and Nickl (2016) one can prove the desired estimates up to a logarithmic factors.

See the complete version of this paper at `https://arxiv.org/pdf/1911.08380.pdf` for details.

ACKNOWLEDGEMENTS

The authors are grateful to Yu. Nesterov and A. Tyurin for fruitful discussions.

REFERENCES

Bach, F. and Levy, K.Y. (2019). A universal algorithm for variational inequalities adaptive to smoothness and noise. In A. Beygelzimer and D. Hsu (eds.), *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, 164–194. PMLR, Phoenix, USA. URL `http://proceedings.mlr.press/v99/bach19a.html`. ArXiv:1902.01637.

Byrd, R.H., Chin, G.M., Nocedal, J., and Wu, Y. (2012). Sample size selection in optimization methods for machine learning. *Mathematical Programming*, 134(1), 127–155.

Deng, Q., Cheng, Y., and Lan, G. (2018). Optimal adaptive and accelerated stochastic gradient descent. *arXiv:1810.00553*.

Devolder, O., Glineur, F., and Nesterov, Y. (2014). First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146(1), 37–75. doi:10.1007/s10107-013-0677-5. URL `http://dx.doi.org/10.1007/s10107-013-0677-5`.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul.), 2121–2159.

Dvinskikh, D., Gorbunov, E., Gasnikov, A., Dvurechensky, P., and Uribe, C.A. (2019). On primal and dual approaches for distributed stochastic convex optimization over networks. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, 7435–7440. doi:10.1109/CDC40024.2019.9029798. ArXiv:1903.09844.

Dvurechensky, P.E., Gasnikov, A.V., and Lagunovskaya, A.A. (2018a). Parallel algorithms and probability of large deviation for stochastic convex optimization problems. *Numerical Analysis and Applications*, 11(1), 33–37. ArXiv:1701.01830.

Dvurechensky, P. (2017). Gradient method with inexact oracle for composite non-convex optimization. *arXiv:1703.09180*.

Dvurechensky, P., Dvinskikh, D., Gasnikov, A., Uribe, C.A., and Nedić, A. (2018b). Decentralize and randomize: Faster algorithm for Wasserstein barycenters. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, NeurIPS 2018, 10783–10793. Curran Associates, Inc. ArXiv:1806.03915.

Dvurechensky, P. and Gasnikov, A. (2016). Stochastic intermediate gradient method for convex problems with stochastic inexact oracle. *Journal of Optimization Theory and Applications*, 171(1), 121–145.

Dvurechensky, P., Gasnikov, A., Stonyakin, F., and Titov, A. (2018c). Generalized Mirror Prox: Solving variational inequalities with monotone operator, inexact oracle, and unknown Hölder parameters. *arXiv:1806.05140*.

Dvurechensky, P., Gasnikov, A., and Tiurin, A. (2017). Randomized similar triangles method: A unifying framework for accelerated randomized optimization methods (coordinate descent, directional search, derivative-free method). *arXiv:1707.08486*.

Dvurechensky, P.E., Gasnikov, A.V., Nurminski, E.A., and Stonyakin, F.S. (2020). *Advances in Low-Memory Subgradient Optimization*, 19–59. Springer International Publishing, Cham. doi: 10.1007/978-3-030-34910-3_2. URL `https://doi.org/10.1007/978-3-030-34910-3_2`. ArXiv:1902.01572.

Friedlander, M.P. and Schmidt, M. (2012). Hybrid deterministic-stochastic methods for data fitting. *SIAM Journal on Scientific Computing*, 34(3), A1380–A1405.

Gasnikov, A.V. and Dvurechensky, P.E. (2016). Stochastic intermediate gradient method for convex optimization problems. *Doklady Mathematics*, 93(2), 148–151.

Gasnikov, A.V., Dvurechensky, P.E., Stonyakin, F.S., and Titov, A.A. (2019). An adaptive proximal method for variational inequalities. *Computational Mathematics and Mathematical Physics*, 59(5), 836–841. doi:10.1134/S0965542519050075. URL `https://doi.org/10.1134/S0965542519050075`.

Gasnikov, A. (2017). Universal gradient descent. *arXiv preprint arXiv:1711.00394*.

Giné, E. and Nickl, R. (2016). *Mathematical foundations of infinite-dimensional statistical models*, volume 40. Cambridge University Press.

Iusem, A.N., Jofré, A., Oliveira, R.I., and Thompson, P. (2019). Variance-based extragradient methods with line search for stochastic variational inequalities. *SIAM Journal on Optimization*, 29(1), 175–206. ArXiv:1703.00262.

Kamzolov, D., Dvurechensky, P., and Gasnikov, A.V. (2020). Universal intermediate gradient method for convex problems with inexact oracle. *Optimization Methods and Software*, 0(0), 1–28. doi:10.1080/10556788.2019.1711079. ArXiv:1712.06036.

Kingma, D. and Ba, J. (2015). Adam: a method for stochastic optimization. *ICLR*.

Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. phd thesis. Technical report, University of Toronto.

Kroshnin, A., Tupitsa, N., Dvinskikh, D., Dvurechensky, P., Gasnikov, A., and Uribe, C. (2019). On the complexity of approximating Wasserstein barycenters. In K. Chaudhuri and R. Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 3530–3540. PMLR, Long Beach, California, USA. ArXiv:1901.08686.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, 2278–2324.

Levy, K.Y., Yurtsever, A., and Cevher, V. (2018). Online adaptive methods, universality and acceleration. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, 6500–6509. Curran Associates, Inc. ArXiv:1809.02864.

Nesterov, Y. (2015). Universal gradient methods for convex optimization problems. *Mathematical Programming*, 152(1), 381–404.

Ogaltsov, A. and Tyurin, A. (2019). Heuristic adaptive fast gradient method in stochastic optimization tasks. *arXiv:1910.04825*.

Stonyakin, F., Gasnikov, A., Tyurin, A., Pasechnyuk, D., Agafonov, A., Dvurechensky, P., Dvinskikh, D., Kroshnin, A., and Piskunova, V. (2019a). Inexact model: A framework for optimization and variational inequalities. *arXiv:1902.00990*.

Stonyakin, F.S., Dvinskikh, D., Dvurechensky, P., Kroshnin, A., Kuznetsova, O., Agafonov, A., Gasnikov, A., Tyurin, A., Uribe, C.A., Pasechnyuk, D., and Artamonov, S. (2019b). Gradient methods for problems with inexact model of the objective. In M. Khachay, Y. Kochetov, and P. Pardalos (eds.), *Mathematical Optimization Theory and Operations Research*, 97–114. Springer International Publishing, Cham. (Lecture Notes in Computer Science; vol. 11548). doi: 10.1007/978-3-030-22629-9_8.

Uribe, C.A., Dvinskikh, D., Dvurechensky, P., Gasnikov, A., and Nedić, A. (2018). Distributed computation of Wasserstein barycenters over networks. In *2018 IEEE Conference on Decision and Control (CDC)*, 6544–6549. ArXiv:1803.02933.

Ward, R., Wu, X., and Bottou, L. (2019). AdaGrad stepsizes: Sharp convergence over nonconvex landscapes. In K. Chaudhuri and R. Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 6677–6686. PMLR, Long Beach, California, USA. URL `http://proceedings.mlr.press/v97/ward19a.html`.

Woodworth, B.E., Wang, J., Smith, A., McMahan, B., and Srebro, N. (2018). Graph oracle models, lower bounds, and gaps for parallel stochastic optimization. In *Advances in neural information processing systems*, 8496–8506.

Yurtsever, A., Tran-Dinh, Q., and Cevher, V. (2015). A universal primal-dual convex optimization framework. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, NIPS'15, 3150–3158. MIT Press, Cambridge, MA, USA.