Adaptive Catalyst for smooth convex optimization ⋆

Anastasiya Ivanova * **Dmitry Grishchenko** **
Alexander Gasnikov *** **Egor Shulgin** ****

* *Moscow Institute of Physics and Technology, Moscow, Russia, National Research University Higher School of Economics, Moscow, Russia, (anastasiya.s.ivanova@phystech.edu)*
** *Université Grenoble Alpes, Grenoble, France*
*** *Moscow Institute of Physics and Technology, Moscow, Russia, Institute for Information Transmission Problems, Moscow, Russia, Caucasus Mathematical Center, Adyghe State University, Russia*
**** *Moscow Institute of Physics and Technology, Moscow, Russia*

Abstract: In 2015 there appears a universal framework *Catalyst* that allows to accelerate almost arbitrary non-accelerated deterministic and randomized algorithms for smooth convex optimization problems Lin et al. (2015). This technique finds a lot of applications in Machine Learning due to the possibility to deal with sum-type target functions. The significant part of the Catalyst approach is accelerated proximal outer gradient method. This method used as an envelope for non-accelerated inner algorithm for the regularized auxiliary problem. One of the main practical problem of this approach is the selection of this regularization parameter. There exists a nice theory for that Lin et al. (2018), but this theory required prior knowledge about the smoothness constant of the target function. In this paper, we propose an adaptive variant of Catalyst that doesn't require such information. In combination with the adaptive inner non-accelerated algorithm, we propose accelerated variants of well-known methods: steepest descent, adaptive coordinate descent, alternating minimization.

Keywords: Adaptive methods, Catalyst, accelerated methods, steepest descent, coordinate descent, alternating minimization

1. INTRODUCTION

One of the main achievement in numerical methods for convex optimization is the development of accelerated methods Nesterov (2018). Until 2015 acceleration schemes for different convex optimization problems seems to be quite different to unify them. But starting from the work Lin et al. (2015) in which universal acceleration technique (*Catalyst*) was proposed, there appear stream of subsequent works Lin et al. (2018); Palaniappan and Bach (2016); Paquette et al. (2017); Kulunchakov and Mairal (2019) that allows spreading Catalyst on monotone variational inequalities, non-convex problems, stochastic optimization problems. In all these works the basic idea is to use an accelerated proximal algorithm as an outer envelope (see Parikh et al. (2014) and references therein) with non-accelerated algorithms for inner auxiliary problems. The main practical drawback of this approach is the requirement to choose a regularization parameter such that the conditional number of the auxiliary problem became $O(1)$. To do that we need to know smoothness parameters of the target that are not typically free available.

Alternative accelerated proximal envelop to Parikh et al. (2014) was proposed in the paper Monteiro and Svaiter (2013). The main difference with the standard accelerated proximal envelops is the adaptability of scheme Monteiro and Svaiter (2013). Note, that this scheme allows also to build (near) optimal tensor (high-order) accelerated methods Gasnikov (2017); Nesterov (2018); Gasnikov et al. (2019a,b); Wilson et al. (2019). That is, the 'acceleration' potential of this scheme seems to be the best known for us for the moment. So the main and rather simple idea of this paper can be formulated briefly as follows: **To develop adaptive Catalyst we replace accelerated proximal envelope with fixed regularization parameter Parikh et al. (2014); Lin et al. (2018) on adaptive accelerated proximal envelope from Monteiro and Svaiter (2013)**.

This replacement described in Section 2.

By using this adaptive accelerated proximal envelop we propose in Section 3 accelerated variant of steepest descent Polyak (1987); Gasnikov (2017) as alternative to A. Nemirovski accelerated steepest descent (see Nesterov et al. (2018); Diakonikolas and Orecchia (2019) and references therein), adaptive accelerated variants of alternating minimization procedures Beck (2017) as an alternative to Diakonikolas and Orecchia (2018); Guminov et al. (2019); Tupitsa et al. (2019) and adaptive accelerated coordinate

⋆ The research of A. Gasnikov was supported by Russian Science Foundation project 18-71-00048 mol-a-ved and was partially supported by Yahoo! Research Faculty Engagement Program.

descent Nesterov (2012). For the last example as far as we know there were no previously complete adaptive accelerated coordinate descent. The most advanced result in this direction is the work Fercoq and Richtárik (2015) that applies only to the problems with increasing smoothness parameter along the iteration process. For example, for the target function like $f(x) = x^4$ this scheme doesn't recognize that smoothness parameters (in particular Lipschitz gradient constant) tends to zero along the iteration process.

In Section 5 we describe numerical experiments with the steepest descent and adaptive coordinate descent.

We hope that the proposed approach allows accelerating not only adaptive on their own procedures, but also many other different non-accelerated non-adaptive randomized schemes by settings on general smoothness parameters of target function that can be difficult to analyze patently Gower et al. (2019); Gorbunov et al. (2019).

2. THE MAIN SCHEME

Let us consider the following minimization problem

$$\min_{y \in Q} f(y), \tag{1}$$

where $f(y)$ is convex function and its gradient is Lipschitz continuous w.r.t. $\|\cdot\|_2$ with the constant L_f:

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L_f \|x - y\|_2.$$

To propose the main scheme of the algorithm we need to define the following functions:

$$F_{L,x}(y) = f(y) + \frac{L}{2}||y - x||_2^2,$$
$$f_L(x) = \min_{y \in Q} F_{L,x}(y) = F_{L,x}(y_L(x)),$$

then the function $F_{L,x}(y)$ is L–strongly convex and its gradient is Lipschitz continuous w.r.t. $\|\cdot\|_2$ with the constant $(L + L_f)$. So, the following inequality holds

$$||\nabla F_{L,x}(y_2) - \nabla F_{L,x}(y_1)||_2 \leq (L + L_f)||y_1 - y_2||_2. \tag{2}$$

Due to this definition, for all $L \geq 0$ we have that $f_L(x) \leq f(x)$ and the convex function $f_L(x)$ has a Lipschitz-continuous gradient with the Lipschitz constant L. Moreover, according to (Polyak, 1987, Theorem 5, ch. 6), since

$$x_\star \in \operatorname*{Argmin}_x f_L(x) = \operatorname*{Argmin}_{x \in Q} f_L(x),$$

we obtain

$$x_\star \in \operatorname*{Argmin}_{x \in Q} f(x), \quad \text{and } f_L(x_\star) = f(x_\star).$$

Thus, instead of the initial problem (1), we can consider the Moreau-Yosida regularized problem

$$\min_{x \in Q} f_L(x). \tag{3}$$

Note that the problem (3) is an ordinary problem of smooth convex optimization. Then the complexity of solving the problem (3) up to the accuracy ε with respect to the function using the Fast Gradient Method (FGM) Nesterov (2018) can be estimated as follows $O\left(\sqrt{\frac{LR^2}{\varepsilon}}\right)$. The 'complexity' means here the number of oracle calls. Each oracle call means calculation of $\nabla f_L(x) = L(x - y_L(x))$, where $y_L(x)$ is the exact solution of the auxiliary problem $\min_{y \in Q} F_{L,x}(y)$.

Note that the smaller the value of the parameter L we choose, the smaller is the number of oracle calls (outer iterations). However, at the same time this increases the complexity of solving the auxiliary problem at each iteration. At the end of this brief introduction to standard accelerated proximal point methods, let us describe the step of ordinary (proximal) gradient descent (for more details see Parikh et al. (2014))

$$x^{k+1} = x^k - \frac{1}{L}\nabla f_L(x) = x^k - \frac{1}{L} \cdot L(x - y_L(x)) = y_L(x).$$

To develop adaptive proximal accelerated envelop we should replace standard FGM Nesterov (2018) on the following adaptive variant of FGM Algorithm 1, introduced by Monteiro and Svaiter (2013) for smooth convex optimization problems. In addition, we also assume that $Q = \mathbb{R}^n$.

Algorithm 1 Monteiro–Svaiter algorithm

1: **Parameters:** $z^0, y^0, A_0 = 0$
2: **for** $k = 0, 1, \ldots, N-1$ **do**
3: Choose L_{k+1} and y^{k+1} such that

$$\|\nabla F_{L_{k+1},x^{k+1}}(y^{k+1})\|_2 \leq \frac{L_{k+1}}{2}\|y^{k+1} - x^{k+1}\|_2,$$

where

$$a_{k+1} = \frac{1/L_{k+1} + \sqrt{1/L_{k+1}^2 + 4A_k/L_{k+1}}}{2},$$
$$A_{k+1} = A_k + a_{k+1},$$
$$x^{k+1} = \frac{A_k}{A_{k+1}}y^k + \frac{a_{k+1}}{A_{k+1}}z^k$$

4: $z^{k+1} = z^k - a_{k+1}\nabla f\left(y^{k+1}\right)$
5: **end for**

The analysis of the algorithm is based on the following theorem.

Theorem 1. (Monteiro and Svaiter, 2013, Theorem 3.6) Let sequence (x^k, y^k, z^k), $k \geq 0$ be generated by Algorithm 1 and define $R := \left\|y^0 - x_\star\right\|_2$. Then, for all $N \geq 0$,

$$\frac{1}{2}\left\|z^N - x_\star\right\|_2^2 + A_N \cdot \left(f\left(y^N\right) - f\left(x_\star\right)\right) + \frac{1}{4}\sum_{k=1}^{N} A_k L_k \left\|y^k - x^k\right\|_2^2 \leq \frac{R^2}{2}, \tag{4}$$

$$f\left(y^N\right) - f\left(x^\star\right) \leq \frac{R^2}{2A_N}, \quad \left\|z^N - x_\star\right\|_2 \leq R, \tag{5}$$

$$\sum_{k=1}^{N} A_k L_k \left\|y^k - x^k\right\|_2^2 \leq 2R^2. \tag{6}$$

We also need the following Lemma.

Lemma 2. (Monteiro and Svaiter, 2013, Lemma 3.7 a)) Let sequences $\{A_k, L_k\}$, $k \geq 0$ be generated by Algorithm 1. Then, for all $N \geq 0$,

$$A_N \geq \frac{1}{4}\left(\sum_{k=1}^{N} \frac{1}{\sqrt{L_k}}\right)^2. \tag{7}$$

Let us define non-accelerated method $\mathcal{M}$ that we will use to solve auxiliary problem

Proposition 3. The convergence rate for the method $\mathcal{M}$ for problem

$$\min_{y \in \mathbb{R}^n} F(y)$$

can be written in the general form as follows (for randomized algorithms, like Algorithm 4, this estimates holds true with high probability)

$$F(y^N) - F(y_\star) = O\left(L_F R^2\right) \min\left\{\frac{C_n}{N}, \exp\left(-\frac{\mu_F N}{C_n L_F}\right)\right\},$$

where $y_\star$ is the solution of the problem, $R = ||y^0 - y_\star||_2$, function F is μ_F–strongly convex and L_F is a constant which characterized smoothness of function F.

Typically $C_n = O(1)$ for the standard full gradient first order methods, $C_n = O(p)$, where p is a number of blocks, for alternating minimization with p blocks and $C_n = O(n)$ for gradient free or coordinate descent methods, where n is dimension of y. See the references in next Remark for details.

Remark 1. Let us clarify what we mean by a constant L_F which characterized smoothness of function F. Typically for the first order methods this is just the Lipschitz constant of gradient F (see, Polyak (1987); De Klerk et al. (2017) for the steepest descent and Karimi et al. (2016); Diakonikolas and Orecchia (2018); Tupitsa et al. (2019) for alternating minimization); for gradient free methods like Algorithm 4 this constant is the average value of the directional smoothness parameters, for gradient free methods see Duchi et al. (2015); Gasnikov et al. (2016); Shamir (2017); Bayandina et al. (2018); Dvurechensky et al. (2018b,a), for coordinate descent methods see Nesterov (2012); Wright (2015); Nesterov and Stich (2017) and for more general situations see Gower et al. (2019).

Remark 2. Note that in proposition 3 the first estimate corresponds to the estimate of the convergence rate of the method $\mathcal{M}$ for convex problems. And the second estimate corresponds to the estimate for strongly convex problems.

Our main goal is to propose a scheme to accelerate methods of this type. But note that we apply our scheme only to degenerate convex problems since it does not take into account the strong convexity of the original problem.

Based on Monteiro–Svaiter accelerated proximal method we propose Algorithm 2.

Now let us prove the main theorem about the convergence rate of the proposed scheme. So, based on the Monteiro–Svaiter Theorem 1 we can introduce the following theorem

Theorem 4. Consider Algorithm 2 with $L_d < L_u$ for solving problem (1), where $Q = \mathbb{R}^n$, with auxiliary (inner) non-accelerated algorithm (method) $\mathcal{M}$ that satisfy Proposition 3 with constants C_n and L_f such that $L_d \leq L_f \leq L_u$.

Algorithm 2 Adaptive Catalyst

Require: Starting point x^0 initial guess $L_0 > 0$; parameters $\alpha > \beta > \gamma > 0$; optimization method $\mathcal{M}$.

for $k = 0, 1, \ldots, N-1$ **do**
  $L_{k+1} = \beta \cdot \min\{\alpha L_k, L_u\}$
  $t = 0$
  repeat
    $t := t + 1$
    $L_{k+1} := \max\{L_{k+1}/\beta, L_d\}$
    Compute

$$a_{k+1} = \frac{1/L_{k+1} + \sqrt{1/L_{k+1}^2 + 4A_k/L_{k+1}}}{2},$$
$$A_{k+1} = A_k + a_{k+1},$$
$$x^{k+1} = \frac{A_k}{A_{k+1}} y^k + \frac{a_{k+1}}{A_{k+1}} z^k.$$

    Compute an approximate solution of the following problem with auxiliary non-accelerated method $\mathcal{M}$

$$y^{k+1} \approx \operatorname*{argmin}_y F_{L_{k+1}, x^{k+1}}(y)$$

    By running $\mathcal{M}$ with starting point x^{k+1} and output point y^{k+1} we wait N_t iterations to fulfill adaptive stopping criteria

$$\|\nabla F_{L_{k+1}, x^{k+1}}(y^{k+1})\|_2 \leq \frac{L_{k+1}}{2}\|y^{k+1} - x^{k+1}\|_2.$$

  until $t > 1$ and $N_t \geq \gamma \cdot N_{t-1}$ or $L_{k+1} = L_d$
  $z^{k+1} = z^k - a_{k+1}\nabla f\left(y^{k+1}\right)$
end for
Output: y^{k+1}

Then the total complexity [1] of the proposed Algorithm 2 with inner method $\mathcal{M}$ is

$$\tilde{O}\left(C_n \cdot \max\left\{\sqrt{\frac{L_u}{L_f}}, \sqrt{\frac{L_f}{L_d}}\right\} \cdot \sqrt{\frac{L_f R^2}{\varepsilon}}\right).$$

Proof.

Note that the Monteiro–Svaiter (M-S) condition

$$\|\nabla F_{L_{k+1}, x^{k+1}}(y^{k+1})\|_2 \leq \frac{L_{k+1}}{2}\|y^{k+1} - x^{k+1}\|_2 \tag{8}$$

instead of the exact solution $y_\star^{k+1} = y_{L_{k+1}}(x^{k+1})$ of the auxiliary problem, for which

$$\|\nabla F_{L_{k+1}, x^{k+1}}(y_\star^{k+1})\|_2 = 0,$$

allows to search inexact solution that satisfies the condition (8).

Since $y_\star^{k+1}$ the solution of the problem $\min\limits_y F_{L_{k+1}, x^{k+1}}(y)$, the $\nabla F_{L_{k+1}, x^{k+1}}(y_\star^{k+1}) = 0$. Then, using inequality (2) we obtain

$$||\nabla F_{L_{k+1}, x^{k+1}}(y^{k+1})||_2 \leq (L_{k+1} + L_f)||y^{k+1} - y_\star^{k+1}||_2 \tag{9}$$

Using the triangle inequality we have

$$||x^{k+1} - y_\star^{k+1}||_2 - ||y^{k+1} - y_\star^{k+1}||_2 \leq ||y^{k+1} - x^{k+1}||_2. \tag{10}$$

Since r.h.s. of the inequality (9) coincide with the r.h.s. of the M-S condition and l.h.s. of the inequality (10) coincide

[1] The number of oracle calls (iterations) of auxiliary method $\mathcal{M}$ that required to find ε solution of (1) in terms of functions value.

with the l.h.s. of the M-S condition up to a multiplicative factor $L_{k+1}/2$, one can conclude that if the inequality

$$||y^{k+1} - y_\star^{k+1}||_2 \leq \frac{L_{k+1}}{3L_{k+1} + 2L_f}||x^{k+1} - y_\star^{k+1}||_2$$

holds, the M-S condition holds too. To solve the auxiliary problem $\min\limits_y F_{L_{k+1},x^{k+1}}(y)$ we use non-accelerated method $\mathcal{M}$. Using proposition (3), we obtain that the convergence rate for these methods can be written as follows

$$\begin{aligned}&F_{L_{k+1},x^{k+1}}(y^N) - F_{L_{k+1},x^{k+1}}(y_\star^{k+1})\\ &=O\left((L_f + L_{k+1})R^2\right)\exp\left(-\frac{L_{k+1}N}{C_n(L_f + L_{k+1})}\right),\end{aligned}$$

where $R^2 = ||x^{k+1} - y_\star^{k+1}||_2$ since x^{k+1} is a starting point.

Since $F_{L_{k+1},x^{k+1}}(\cdot)$ is L_{k+1}-strongly convex function, the following inequality holds Nesterov (2018)

$$\frac{L_{k+1}}{2}||y^{k+1} - y_\star^{k+1}||_2^2 \leq F_{L_{k+1},x^{k+1}}(y^N) - F_{L_{k+1},x^{k+1}}(y_\star^{k+1}).$$

Thus,

$$\begin{aligned}&||y^{k+1} - y_\star^{k+1}||_2\\ &\leq O\left(\sqrt{\frac{(L_f + L_{k+1})R^2}{L_{k+1}}}\right)\exp\left(-\frac{L_{k+1}N}{2C_n(L_f + L_{k+1})}\right).\end{aligned}$$

From this, we obtain that the complexity of solving the auxiliary problem is

$$T = \tilde{O}\left(C_n\frac{(L_{k+1} + L_f)}{L_{k+1}}\right)$$

Note, that $\tilde{O}(\cdot)$ means the same as $O(\cdot)$ up to a logarithmic factor. Since that we can consider T to be the estimate that include total complexity of auxiliary problem including all inner restarts on L_{k+1}.

Substituting inequality (7) into estimation (6) we obtain

$$f(y^N) - f(x_\star) \leq \frac{2R^2}{\left(\sum\limits_{k=1}^{N}\frac{1}{\sqrt{L_k}}\right)^2}.$$

Since the complexity of the auxiliary problem is T we can estimate the complexity in the worst two cases as follows:

- If all $L_{k+1} = L_d \leq L_f$ (at each iteration we estimate the regularization parameter as lower bound), then $\frac{(L_{k+1}+L_f)}{L_{k+1}} \approx \frac{L_f}{L_{k+1}}$ and total complexity is
$$\tilde{O}\left(C_n\frac{L_f}{L_d}\sqrt{\frac{L_dR^2}{\varepsilon}}\right) = \tilde{O}\left(C_n\sqrt{\frac{L_f}{L_d}}\cdot\sqrt{\frac{L_fR^2}{\varepsilon}}\right).$$
- If all $L_{k+1} = L_u \geq L_f$ (at each iteration we estimate the regularization parameter as upper bound), then $\frac{(L_{k+1}+L_f)}{L_{k+1}} \approx 1$ and total complexity is
$$\tilde{O}\left(C_n\sqrt{\frac{L_uR^2}{\varepsilon}}\right) = \tilde{O}\left(C_n\sqrt{\frac{L_u}{L_f}}\cdot\sqrt{\frac{L_fR^2}{\varepsilon}}\right).$$

Then, using these two estimations we obtain the result of the theorem.

Note that this result shows that such a procedure will works not worse than standard Catalyst Lin et al. (2015, 2018) up to a factor $\tilde{O}\left(\max\left\{\sqrt{\frac{L_u}{L_f}}, \sqrt{\frac{L_f}{L_d}}\right\}\right)$ independent on the stopping criteria in the restarts on L_{k+1}.

Since the complexity of solving the auxiliary problem is proportional to $\frac{(L_{k+1}+L_f)C_n}{L_{k+1}}$, when we reduce the parameter L_{k+1} so that $L_{k+1} < L_f$ the complexity of solving an auxiliary problem became growth exponentially. Therefore, as the stopping criterion of the inner method, we select the number of iterations N_t compared to the number of iterations N_{t-1} at the previous restart $t-1$. This means that if $N_t \leq \gamma N_{t-1}$ then the complexity begins to grow exponentially and it is necessary to go to the next iteration of the external method. By using such adaptive rule we try to recognize the best possible value of $L_{k+1} \simeq L_f$. The last facts is a basis of standard Catalyst approach Lin et al. (2015, 2018) an have a very simple explanation. To minimize the total complexity we should take parameter $L_{k+1} \equiv L$ such that

$$\min_L \sqrt{\frac{LR^2}{\varepsilon}}\cdot\tilde{O}\left(\frac{L_f + L}{L}\right).$$

This lead us to $L_{k+1} \simeq L_f$.

3. APPLICATIONS

In this section we present few examples of algorithms that we consider as inner solvers. All of them has adaptive structure, so it makes little sense to use Catalyst algorithm to accelerate them.

3.1 Steepest descent

Consider the following problem

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

where $f(x)$ is a L_f–smooth convex function (its gradient is Lipschitz continuous w.r.t. $\|\cdot\|_2$ with the constant L_f).

To solve this problem let us consider general gradient descent update rule

$$x^{k+1} = x^k - h_k\nabla f(x^k).$$

In Polyak (1987) it was proposed an adaptive way to select h_k as following (see also De Klerk et al. (2017) for precise rates of convergence)

$$h_k = \operatorname*{argmin}_{h\in\mathbb{R}} f\left\{x^k - h\nabla f(x^k)\right\}.$$

Algorithm 3 Steepest descent

Require: Starting point x^0.
Ensure: x^k
1: **for** $k \geqslant 0$ **do**
2: Choose $h_k = \operatorname{argmin}_{h\in\mathbb{R}} f\left\{x^k - h\nabla f(x^k)\right\}$
3: Set $x^{k+1} = x^k - h_k\nabla f(x^k)$
4: **end for**

In contrast with the standard selection $h_k \equiv \frac{1}{L_f}$ for L-smooth functions f, in this method there is no need to know smoothness constant of the function. It allows to use this method for the smooth functions f when L_f is unknown (or expensive to compute) or when the global L_f is much bigger than the local ones along the trajectory.

On the other hand, as far as we concern there is no direct acceleration of steepest descent algorithm. Moreover, it is hard to use Catalyst with it as far as acceleration happens if L_k (κ in Catalyst article notations) is selected with respect to L_f and the scheme does not support adaptivity out of the box. Even if global L_f is known the local smoothness constant could be significantly different from it that will lead to worse speed of convergence.

Note that for Algorithm 3 the proposition 3 holds with $C_n = O(1)$ and L_f is the Lipschitz constant of the gradient of function f.

3.2 Random Adaptive Coordinate Decent Method

Consider the following unconstrained problem

$$\min_{x \in \mathbb{R}^n} f(x).$$

Now we assume directional smoothness for f, that is there exists $\beta_1, \ldots, \beta_n$ such that for any $x \in \mathbb{R}^n, u \in \mathbb{R}$

$$|\nabla_i f\left(x + u e_i\right) - \nabla_i f(x)| \leq \beta_i |u|, \quad i = 1, \ldots, n,$$

where $\nabla_i f(x) = \partial f(x)/\partial x_i$. For twice differentiable f it equals to $(\nabla^2 f(x))_{i,i} \leq \beta_i$. Due to the fact that we consider the situation when smoothness constants are not known we use such dynamic adjustment scheme from Nesterov (2012); Wright (2015).

Algorithm 4 RACDM

Require: Starting point x^0;
lower bounds $\hat{\beta}_i := \beta_i^0 \in (0, \beta_i], i = 1, \ldots, n$
Ensure: x^k
1: **for** $k \geqslant 0$ **do**
2: Sample $i_k \sim \mathcal{U}[1, \ldots, n]$
3: Set $x^{k+1} = x^k - \hat{\beta}_{i_k}^{-1} \cdot \nabla_{i_k} f\left(x^k\right) \cdot e_{i_k}$
4: **While** $\nabla_{i_k} f(x^k) \cdot \nabla_{i_k} f(x^{k+1}) < 0$ **do**
$\left\{\hat{\beta}_{i_k} = 2\hat{\beta}_{i_k}, \quad x^{k+1} = x^k - \hat{\beta}_{i_k}^{-1} \cdot \nabla_{i_k} f\left(x^k\right) \cdot e_{i_k}\right\}$
5: Set $\beta_{i_k} = \frac{1}{2}\beta_{i_k}$
6: **end for**

Note that for Algorithm 4 the proposition 3 holds with $C_n = O(n)$ (for $x \in \mathbb{R}^n$) and $L_f = \overline{L}_f := \frac{1}{n}\sum\limits_{i=1}^{n} \beta_i$ (the average value of the directional smoothness parameters).

3.3 Alternating Minimization

Consider the following problem

$$\min_{x \in Q \subseteq E} f(x),$$

where $f(x)$ is a L_f–smooth convex function (its gradient is Lipschitz continuous w.r.t. $\|\cdot\|_2$ with the constant L_f), $Q = \otimes_{i=1}^{p} Q_i \subseteq E$, with $Q_i \subseteq E_i$, $i = 1, ..., p$ being closed convex sets.

For the general case of number of blocks $p \geqslant 2$ the Alternating Minimization algorithm may be written as Algorithm 5. There are multiple common block selection rules, such as the cyclic rule or the Gauss–Southwell rule Karimi et al. (2016); Beck (2017); Diakonikolas and Orecchia (2018); Tupitsa et al. (2019).

Algorithm 5 Alternating Minimization

Require: Starting point x^0.
Ensure: x^k
1: **for** $k \geqslant 0$ **do**
2: Choose $i \in 1, \ldots, p$
3: Set $x^{k+1} = \underset{x \in S_i(x^k)}{\operatorname{argmin}} f(x)$
4: **end for**

Note that for Algorithm 5 the proposition 3 holds with $C_n = O(p)$ (p – number of blocks) and L_f is the Lipschitz constant of the gradient of function f.

4. THEORETICAL GUARANTEES

Let us present the table that establishes the comparison of rates of convergence for the above algorithms before and after acceleration via Algorithm 2. In non-accelerated case these algorithms apply to the convex but non-strongly convex problem, therefore, we use estimates for the convex case from proposition 3. But in the case of acceleration of these methods, we apply them to a regularized function which is strongly convex.

	non-acc	M-S acc
SD	$\frac{L_f R^2}{\varepsilon}$	$\max\left(\sqrt{\frac{L_u}{L_f}}, \sqrt{\frac{L_f}{L_d}}\right)\sqrt{\frac{L_f R^2}{\varepsilon}}$
RACDM	$n \cdot \frac{\overline{L}_f R^2}{\varepsilon}$	$n \cdot \max\left(\sqrt{\frac{L_u}{\overline{L}_f}}, \sqrt{\frac{\overline{L}_f}{L_d}}\right)\sqrt{\frac{\overline{L}_f R^2}{\varepsilon}}$
AM	$p \cdot \frac{L_f R^2}{\varepsilon}$	$p \cdot \max\left(\sqrt{\frac{L_u}{L_f}}, \sqrt{\frac{L_f}{L_d}}\right)\sqrt{\frac{L_f R^2}{\varepsilon}}$

5. EXPERIMENTS

In this section, we perform experiments for justifying acceleration of the aforementioned methods in practice.

5.1 RADCM Acceleration

Let us consider a simple problem of quadratic optimization

$$\min_{x \in \mathbb{R}^n} \quad f(x) = \frac{1}{2} x^\top A x, \tag{11}$$

where matrix $A = S^\top D S$ is synthetically generated in such a way that S is a random orthogonal matrix. The elements of diagonal matrix D are sampled from standard uniform distribution $\mathcal{U}(0, 1)$ and one random D_{ii} is assigned to zero to guarantee that the smallest eigenvalue of the resulting matrix A is smaller than 10^{-15} and thus the optimization problem is convex but not strongly-convex (up to machine precision). [2]

[2] Frankly speaking, for such objective functions we observe that non-accelerated gradient descent based algorithms converge with linear rate, because of the quadratic nature of the problem and specificity of spectrum. Since $n = 100$ in these experiments we typically have that the next eigenvalue after zero is about 0.01. This value determined the real rate of convergence.

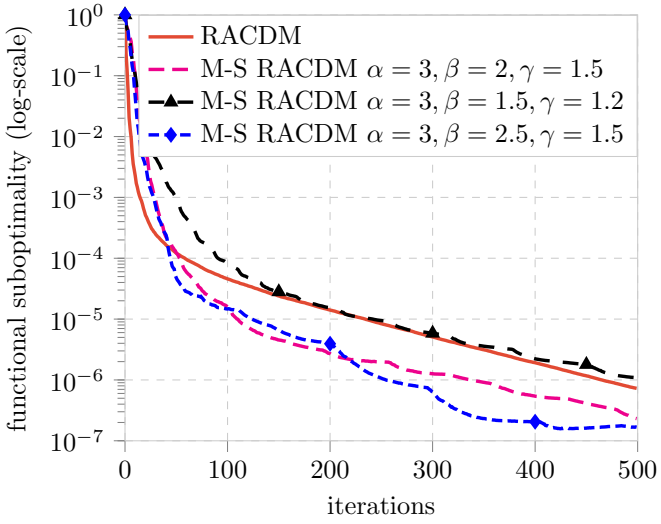

Fig. 1. Quadratic problem (11) with synthetically generated data.

In Figure 1 we compare the performance of the method 4 and its M-S accelerated version with different sets of parameters (α, β, γ) for problem 11. We evaluate the functional suboptimality measure $f(y) - f^\star$ $(f^\star = 0)$. For the horizontal axis we use number of partial derivative evaluations divided by dimensionality n of the problem. Our warm start strategy includes running inner method from the last point y_k and with estimates $\hat{\beta}_i$ of smoothness constants obtained from the previous iteration. The initial points $y_0 = z_0$ are sampled from the standard uniform distribution $\mathcal{U}(0, 1)$. L_0 was initialized as $1.6L_f$ and $L_d = 0.005L_f, L_u = 10L_f, \beta_i^0 = 1/L_0$. We observe that clear acceleration can be achieved not for all sets of parameters. Concretely, both β and γ affected convergence as one can see from the plot. Besides, we find out that for higher accuracy the proposed method can show unstable performance.

Note, that we can obtain provable acceleration by the proposed Adaptive Catalyst procedure described in Algorithm 2 only for convex problems. For strongly convex problems, this is no longer true either in theory or in our experiments. The reason is that the proposed M-S accelerated envelop doesn't take in to account possible strong convexity Gasnikov (2017). Moreover, as far as we know, this is still an open problem, to propose a fully adaptive accelerated algorithm for strongly convex problems. The problem is in the strong convexity parameter. In practice, we met this problem in different places. For example, when we choose the restart parameter for conjugate gradient methods or try to propose accelerated (fast) gradient methods that do not require any information about strong convexity parameter but know all other characteristics of the problem Gasnikov (2017).

5.2 Steepest descent

Let us consider logistic loss minimization problem

$$\min_{x \in \mathbb{R}^n} \quad f(x) = \frac{1}{m} \sum_{j=1}^{m} \log(1 + \exp(-y_j z_j^\top x)) \qquad (12)$$

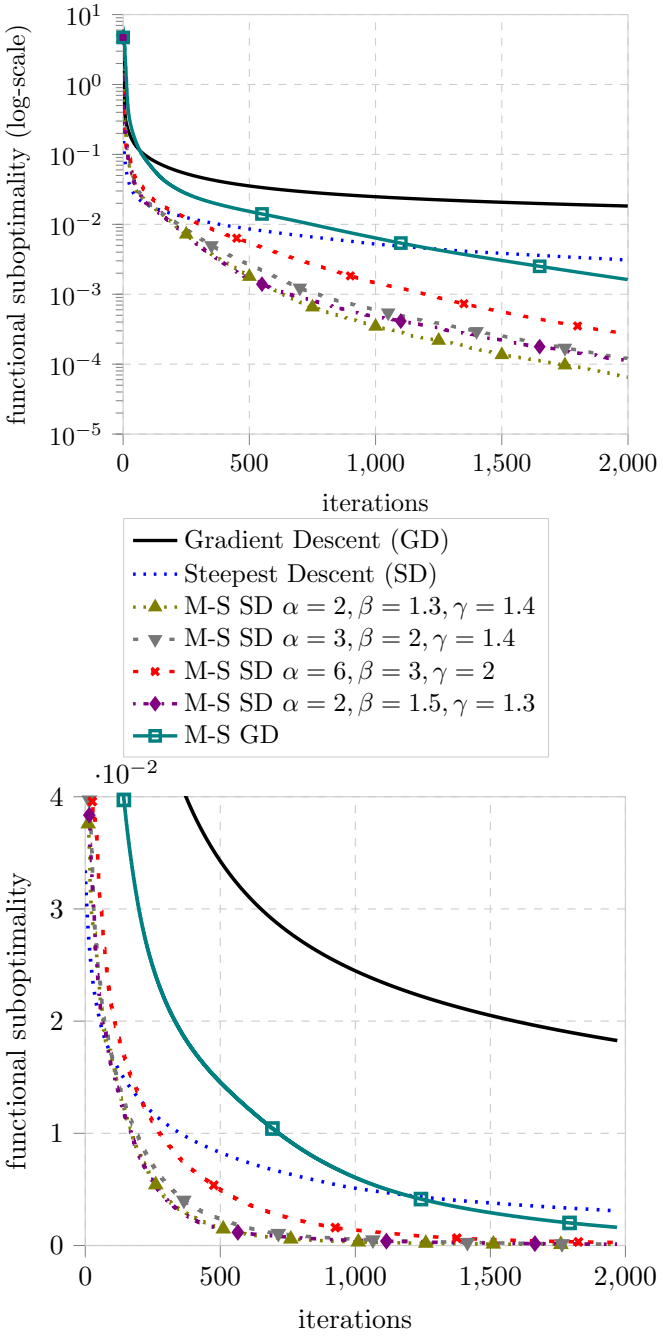

Fig. 2. Logistic regression (12) with a1a dataset from LIBSVM repository.

with *a1a* dataset from LIBSVM Chang and Lin (2011) repository. In contrast with the function considered in previous case, logistic regression converges with sub-linear rate like general non-strongly function.

We present our experimental results for *gradient descent*, *steepest descent* (Algorithm 3), and their accelerated via Algorithm 2 versions. In addition, we made several runs of *accelerated steepest descent* to show the dependence of the algorithm on (α, β, γ). [3]

[3] For all runs with steepest descent we used $L_d = 0.01L_f$ and $L_u = L_f$, where L_f is a real Lipschitz constant of ∇f.

In Figure 2 we present functional suboptimality (not only in points y^k) vs aggregated amount of gradient computations (oracle calls) in two scales: logarithmic (upper plot) and usual scale (to see $1/k^2$ vs $1/k$ rates difference).

As we could see from both plots, acceleration happens when M-S acceleration scheme is used together with steepest descent algorithm but is highly dependent on them. For instance, big α and β makes it harder to algorithm to adapt to the current "optimal" value of L_k that makes algorithm slower. Second, selecting big γ is not reasonable too as far as it corresponds to the big fluctuation of L_k during every restart. Moreover, selecting α and β close to each other also tends to slow down the convergence process. Finally, we add comparison with accelerated via Algorithm 2 gradient descent to show, that accelerated adaptive method works better than non-adaptive one with the same arguments as it for the non-accelerated case.

ACKNOWLEDGEMENTS

We would like to thank Pavel Dvurechensky (WIAS, Berlin) and Peter Richtarik (KAUST) for useful remarks.

REFERENCES

Bayandina, A., Gasnikov, A., and Lagunovskaya, A. (2018). Gradient-free two-points optimal method for non smooth stochastic convex optimization problem with additional small noise. *Automation and remote control*, 79(7). ArXiv:1701.03821.

Beck, A. (2017). *First-order methods in optimization*, volume 25. SIAM.

Chang, C.C. and Lin, C.J. (2011). Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3), 27.

De Klerk, E., Glineur, F., and Taylor, A.B. (2017). On the worst-case complexity of the gradient method with exact line search for smooth strongly convex functions. *Optimization Letters*, 11(7), 1185–1199.

Diakonikolas, J. and Orecchia, L. (2018). Alternating randomized block coordinate descent. *arXiv preprint arXiv:1805.09185*.

Diakonikolas, J. and Orecchia, L. (2019). Conjugate gradients and accelerated methods unified: The approximate duality gap view. *arXiv preprint arXiv:1907.00289*.

Duchi, J.C., Jordan, M.I., Wainwright, M.J., and Wibisono, A. (2015). Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Trans. Information Theory*, 61(5), 2788–2806.

Dvurechensky, P., Gasnikov, A., and Gorbunov, E. (2018a). An accelerated directional derivative method for smooth stochastic convex optimization. *arXiv:1804.02394*.

Dvurechensky, P., Gasnikov, A., and Gorbunov, E. (2018b). An accelerated method for derivative-free smooth stochastic convex optimization. *arXiv:1802.09022*.

Fercoq, O. and Richtárik, P. (2015). Accelerated, parallel, and proximal coordinate descent. *SIAM Journal on Optimization*, 25(4), 1997–2023.

Gasnikov, A.V., Lagunovskaya, A.A., Usmanova, I.N., and Fedorenko, F.A. (2016). Gradient-free proximal methods with inexact oracle for convex stochastic nonsmooth optimization problems on the simplex. *Automation and Remote Control*, 77(11), 2018–2034. doi:10.1134/S0005117916110114. URL `http://dx.doi.org/10.1134/S0005117916110114`. ArXiv:1412.3890.

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

Gasnikov, A., Dvurechensky, P., Gorbunov, E., Vorontsova, E., Selikhanovych, D., Uribe, C.A., Jiang, B., Wang, H., Zhang, S., Bubeck, S., et al. (2019a). Near optimal methods for minimizing convex functions with lipschitz p-th derivatives. In *Conference on Learning Theory*, 1392–1393.

Gasnikov, A., Gorbunov, E., Kovalev, D., Mokhammed, A., and Chernousova, E. (2019b). Reachability of optimal convergence rate estimates for high-order numerical convex optimization methods. In *Doklady Mathematics*, volume 99, 91–94. Springer.

Gorbunov, E., Hanzely, F., and Richtarik, P. (2019). A unified theory of sgd: Variance reduction, sampling, quantization and coordinate descent.

Gower, R.M., Loizou, N., Qian, X., Sailanbayev, A., Shulgin, E., and Richtárik, P. (2019). Sgd: General analysis and improved rates. *arXiv preprint arXiv:1901.09401*.

Guminov, S., Dvurechensky, P., and Gasnikov, A. (2019). Accelerated alternating minimization. *arXiv preprint arXiv:1906.03622*.

Karimi, H., Nutini, J., and Schmidt, M. (2016). Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 795–811. Springer.

Kulunchakov, A. and Mairal, J. (2019). A generic acceleration framework for stochastic composite optimization. *arXiv preprint arXiv:1906.01164*.

Lin, H., Mairal, J., and Harchaoui, Z. (2015). A universal catalyst for first-order optimization. In *Advances in neural information processing systems*, 3384–3392.

Lin, H., Mairal, J., and Harchaoui, Z. (2018). Catalyst acceleration for first-order convex optimization: from theory to practice. *arXiv preprint arXiv:1712.05654*.

Monteiro, R.D. and Svaiter, B.F. (2013). An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. *SIAM Journal on Optimization*, 23(2), 1092–1125.

Nesterov, Y. (2012). Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2), 341–362.

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

Nesterov, Y., Gasnikov, A., Guminov, S., and Dvurechensky, P. (2018). Primal-dual accelerated gradient descent with line search for convex and nonconvex optimization problems. *arXiv preprint arXiv:1809.05895*.

Nesterov, Y. and Stich, S.U. (2017). Efficiency of the accelerated coordinate descent method on structured optimization problems. *SIAM Journal on Optimization*, 27(1), 110–123.

Palaniappan, B. and Bach, F. (2016). Stochastic variance reduction methods for saddle-point problems. In *Advances in Neural Information Processing Systems*, 1416–

1424.

Paquette, C., Lin, H., Drusvyatskiy, D., Mairal, J., and Harchaoui, Z. (2017). Catalyst acceleration for gradient-based non-convex optimization. *arXiv preprint arXiv:1703.10993*.

Parikh, N., Boyd, S., et al. (2014). Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3), 127–239.

Polyak, B.T. (1987). *Introduction to optimization*. No. 04; QA402. 5, P6.

Shamir, O. (2017). An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *Journal of Machine Learning Research*, 18, 52:1–52:11.

Tupitsa, N., Dvurechensky, P., and Gasnikov, A. (2019). Alternating minimization methods for strongly convex optimization. *arXiv preprint arXiv:1911.08987*.

Wilson, A.C., Mackey, L., and Wibisono, A. (2019). Accelerating rescaled gradient descent: Fast optimization of smooth functions. In *Advances in Neural Information Processing Systems*, 13533–13543.

Wright, S.J. (2015). Coordinate descent algorithms. *Mathematical Programming*, 151(1), 3–34.