Optimal Tensor Methods in Smooth Convex and Uniformly Convex Optimization

Alexander Gasnikov GASNIKOV@YANDEX.RU
Moscow Institute of Physics and Technology, Institute for Information Transmission Problems, National Research University Higher School of Economics

Pavel Dvurechensky PAVEL.DVURECHENSKY@GMAIL.COM
Weierstrass Institute for Applied Analysis and Stochastics, Institute for Information Transmission Problems

Eduard Gorbunov EDUARD.GORBUNOV@PHYSTECH.EDU
Moscow Institute of Physics and Technology

Evgeniya Vorontsova VORONTSOVAEA@GMAIL.COM
Far Eastern Federal University

Daniil Selikhanovych SELIHANOVICH.DO@PHYSTECH.EDU
Moscow Institute of Physics and Technology, Institute for Information Transmission Problems

César A. Uribe CAURIBE@MIT.EDU
Massachusetts Institute of Technology

Editors: Alina Beygelzimer and Daniel Hsu

Abstract

We consider convex optimization problems with the objective function having Lipshitz-continuous p-th order derivative, where $p \geq 1$. We propose a new tensor method, which closes the gap between the lower $\Omega\left(\varepsilon^{-\frac{2}{3p+1}}\right)$ and upper $O\left(\varepsilon^{-\frac{1}{p+1}}\right)$ iteration complexity bounds for this class of optimization problems. We also consider uniformly convex functions, and show how the proposed method can be accelerated under this additional assumption. Moreover, we introduce a p-th order condition number which naturally arises in the complexity analysis of tensor methods under this assumption. Finally, we make a numerical study of the proposed optimal method and show that in practice it is faster than the best known accelerated tensor method. We also compare the performance of tensor methods for $p = 2$ and $p = 3$ and show that the 3rd-order method is superior to the 2nd-order method in practice.

Keywords: Convex optimization, unconstrained minimization, tensor methods, worst-case complexity, global complexity bounds, condition number

1. Introduction

In this paper, we consider the unconstrained convex optimization problem

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

where f has p-th Lipschitz-continuous derivative with constant M_p. For $p = 1$, first-order methods are commonly used to solve this problem, i.e., gradient descent. The lower bound for the complexity of these methods was proposed in (Nemirovsky and Yudin, 1983; Nesterov, 2004), and an optimal method was introduced in (Nesterov, 1983). The case of $p = 2$, i.e., Newton-type methods, was well understood only recently. A nearly optimal method was proposed in (Nesterov, 2008), an optimal method was proposed in (Monteiro and Svaiter, 2013), and a lower bound was obtained in (Agarwal and Hazan, 2018; Arjevani et al., 2018).

The idea of using higher order derivatives (starting from $p \geq 3$) in optimization is known at least since 1970's, see Hoffmann and Kornstaedt (1978). Recently this direction of research became of interest from the point of view of complexity bounds. In the unpublished preprint Baes (2009), extending the estimating functions technique of Nesterov (2004), proposes accelerated high-order (tensor) methods for convex problems with complexity $O\left(\left(M_p R^{p+1}/\varepsilon\right)^{\frac{1}{p+1}}\right)$, where $p \geq 1$, ε is the accuracy of the obtained solution $\hat{x}$, i.e., $f(\hat{x}) - f^* \leq \varepsilon$, M_p is the Lipschitz constant of the p-th derivative, and R is an estimate for the distance between a starting point and the closest solution. Nevertheless, the author doubts that the obtained methods are implementable since the auxiliary problem on each iteration is possibly non-convex. Agarwal and Hazan (2018); Arjevani et al. (2018) construct lower complexity bounds $\Omega\left(\left(M_p R^{p+1}/\varepsilon\right)^{\frac{2}{5p+1}}\right)$ and $\Omega\left(\left(M_p R^{p+1}/\varepsilon\right)^{\frac{2}{3p+1}}\right)$ respectively for the case f having Lipschitz p-th derivative and conjecture that the upper bound can be improved. Nesterov (2018a) proposes implementable tensor methods showing that an appropriately regularized Taylor expansion of a convex function is again a convex function, thus making the auxiliary problems on each iteration of the tensor methods tractable. The author also provides an accelerated scheme with complexity bound $O\left(\left(M_p R^{p+1}/\varepsilon\right)^{\frac{1}{p+1}}\right)$, shows that the complexity of each iteration for $p = 3$ is of the same order as for the case $p = 2$, and conjectures the existence of an optimal[1] scheme with complexity bound $O\left(\left(M_p R^{p+1}/\varepsilon\right)^{\frac{2}{3p+1}}\right)$.

The optimal method for the case $p = 1$ has complexity $O\left(\left(M_1 R^2/\varepsilon\right)^{\frac{1}{2}}\right)$ (Nesterov, 1983) and for $p = 2$ has the complexity $O\left(\left(M_2 R^3/\varepsilon\right)^{\frac{2}{7}}\right)$ (Monteiro and Svaiter, 2013), but the question of existence of optimal methods for $p \geq 3$ remains open. In this paper we extend the framework of Monteiro and Svaiter (2013) and propose optimal tensor methods for all $p \geq 1$. Our approach is also based on regularized Taylor step of Nesterov (2018a), thus, our optimal method for $p = 2$ is different from Monteiro and Svaiter (2013).

We also consider problem (1) under the additional assumption that f is uniformly convex on the convex bounded set Q, i.e., there exist $2 \leq q \leq p+1$ and $\sigma_q > 0$ s.t.

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

One can show that iterates of Algorithm 3 lie in the ball $B_{R_1}(x^*)$ with radius $R_1 = O(R)$ and center at the solution x^* (see Bubeck et al. (2018) for the details). In this case one can consider uniformly convex functions on $Q = B_{R_1}(x^*)$.

Under this additional assumption, we show, how the restart technique can be applied to accelerate our method to obtain complexity

$$O\left(\left(\frac{M_p}{\sigma_{p+1}}\right)^{\frac{2}{3p+1}} \log_2 \frac{\Delta_0}{\varepsilon}\right), q = p+1; \quad O\left(\left(\frac{M_p (\Delta_0)^{\frac{p+1-q}{q}}}{{\sigma_q}^{\frac{p+1}{q}}}\right)^{\frac{2}{3p+1}} + \log_2 \frac{\Delta_0}{\varepsilon}\right), q < p+1,$$

1. We say that the method is optimal if the dependence on M_p, R_p and ε of the complexity bound can not be improved. However, numerical constants could depend on p. We neglect such dependencies.

where $f(x_0) - f^* \leq \Delta_0$. This bound suggests a natural generalization of first- and second-order condition number (Nesterov, 2008). If f is such that $q = p+1$, then the complexity of our algorithm depends only logarithmically on the starting point and is proportional to

$$(\gamma_p)^{\frac{2}{3p+1}} ,$$

where $\gamma_p = M_p/\sigma_{p+1}$ is the p-th order condition number. Nemirovsky and Yudin (1983); Nesterov (2004) and Arjevani et al. (2018) propose lower bounds for particular cases of strongly convex functions (i.e., $q = 2$) with $p = 1$ and $p = 2$ respectively. Our upper bounds match them. After this paper was accepted, the paper Doikov and Nesterov (2019) appeared on arXiv. This paper considers cubic regularization of Newton method for uniformly convex functions with Hölder-continuoius Hessian. There is an overlap of this class of functions with the considered here case of $p = 2$ and $q = 3$, for which our complexity bound is better.

As a related work, we also mention Birgin et al. (2017); Cartis et al. (2018), who study complexity bounds for tensor methods for finding approximate stationary points with the main focus on non-convex optimization, which we do not consider in our work. Also the work in (Wibisono et al., 2016) considers tensor methods from the variational perspective and obtains similar bounds to those in Baes (2009). The nature of acceleration of higher-order methods is discussed in Wilson et al. (2019).

The first version of this paper appeared in arXiv on September 2, 2018. In December 2018, two months after that, Jiang et al. (2018); Bubeck et al. (2018) proposed an algorithm, which is very similar to our Algorithm 1. Unlike them, we also analyze the case of uniformly convex functions and propose an algorithm, which is faster in this case in terms of iteration complexity, see our Algorithm 3. Moreover, we are the first to make a numerical study of tensor methods for $p = 3$ and show that they work in practice.

Our contributions.

- We propose a new optimal tensor method and analyze its iteration complexity.
- We generalize this method for the case of uniformly convex objectives and propose a definition of p-th order condition number.
- We make a numerical study of the proposed method and show that our optimal method is faster than accelerated tensor method Nesterov (2018a) in practice. We also compare the performance of tensor methods for $p = 2$ and $p = 3$ and show that the 3rd-order method is superior to the 2nd-order method in practice.

We notice that one can consider the class of convex functions with of Hölder continuous higher-order derivatives as in Grapiglia and Nesterov (2019) and generalize results from our paper to this setting to obtain better complexity bounds than in Grapiglia and Nesterov (2019). We leave this for future work.

Notations and generalities. For $p \geq 1$, we denote by $\nabla^p f(x)[h_1, ..., h_p]$ the directional derivative of function f at x along directions $h_i \in \mathbb{R}^n$, $i = 1, ..., p$. $\nabla^p f(x)[h_1, ..., h_p]$ is symmetric p-linear form and its norm is defined as

$$\|\nabla^p f(x)\|_2 = \max_{h_1,...,h_p \in \mathbb{R}^n} \{\nabla^p f(x)[h_1, ..., h_p] : \|h_i\|_2 \leq 1, i = 1, ..., p\}$$

or equivalently

$$\|\nabla^p f(x)\|_2 = \max_{h\in\mathbb{R}^n}\{|\nabla^p f(x)[h,...,h]| : \|h\|_2 \le 1, i = 1,...,p\}.$$

Here, for simplicity, $\|\cdot\|_2$ is standard Euclidean norm, but our algorithm and derivations can be generalized for the Euclidean norm given by general a positive semi-definite matrix B. We consider convex, p times differentiable on $\mathbb{R}$ functions satisfying Lipschitz condition for p-th derivative

$$\|\nabla^p f(x) - \nabla^p f(y)\|_2 \le M_p\|x-y\|_2, x, y \in \mathbb{R}^n. \tag{2}$$

2. Optimal Tensor Method

Given a function f, numbers $p \ge 1$ and $M \ge 0$, define

$$T^f_{p,M}(x) \in \text{Arg}\min_{y\in\mathbb{R}^n}\left\{\sum_{r=0}^{p}\frac{1}{r!}\nabla^r f(x)\underbrace{[y-x,...,y-x]}_{r} + \frac{M}{(p+1)!}\|y-x\|_2^{p+1}\right\}, \tag{3}$$

and given a number $L \ge 0$ and point $z \in \mathbb{R}^n$, we define

$$F_{L,z}(x) \triangleq f(x) + \frac{L}{2}\|x-z\|_2^2. \tag{4}$$

Theorem 1 *Let sequence (x^k, y^k, u^k), $k \ge 0$ be generated by Algorithm 1. Then*

$$f(y^N) - f^* \le \frac{cM_p\|y^0 - x_*\|_2^{p+1}}{N^{\frac{3p+1}{2}}}, \quad \textit{and} \quad c = \frac{2^{\frac{3(p+1)^2+4}{4}}(p+1)}{p!}.$$

Note that this bound allows to obtain an $O\left(\left(M_pR^{p+1}/\varepsilon\right)^{\frac{2}{3p+1}}\right)$ iteration complexity. The implementability and cost of each iteration is discussed below in Section 2.3. The proof of Theorem 1 is based on the framework of Monteiro and Svaiter (2013), which is presented in the next subsection.

Algorithm 1 Optimal Tensor Method

Input: u_0, y_0 — starting points; N — iteration number; $A_0 = 0$

Output: y^N

1: **for** $k = 0, 1, 2, \ldots, N-1$ **do**

2: Choose L_k such that

$$\frac{1}{2} \le \frac{2(p+1)M_p}{p!L_k}\|y^{k+1} - x^k\|_2^{p-1} \le 1, \tag{5}$$

where

$$a_{k+1} = \frac{1/L_k + \sqrt{1/L_k^2 + 4A_k/L_k}}{2}, \quad A_{k+1} = A_k + a_{k+1}, \quad \{\text{note that } L_k a_k^2 = A_{k+1}\}$$

$$x^k = \frac{A_k}{A_{k+1}}y^k + \frac{a_{k+1}}{A_{k+1}}u^k, \quad y^{k+1} = T^{F_{L_k,x^k}}_{p,pM_p}(x^k).$$

3: $u^{k+1} = u^k - a_{k+1}\nabla f(y^{k+1})$

4: **end for**

5: **return** y^N

2.1. Accelerated hybrid proximal extragradient method

Monteiro and Svaiter (2013) introduced Algorithm 2 for convex optimization problems. To find y^{k+1} on each iteration, the authors use gradient type method for the case $p = 1$ and a trust region Newton-type method for the case $p = 2$. Their analysis of the algorithm is based on the following Theorem.

Theorem 2 ((Monteiro and Svaiter, 2013, Theorem 3.6)) *Let sequence (x^k, y^k, u^k), $k \geq 0$ be generated by Algorithm 2 and define $R := \left\|y^0 - x_*\right\|_2$. Then, for all $N \geq 0$,*

$$\frac{1}{2}\left\|u^N - x_*\right\|_2^2 + A_N \cdot \left(f\left(y^N\right) - f\left(x_*\right)\right) + \frac{1}{4}\sum_{k=1}^{N} A_k L_{k-1}\left\|y^k - x^{k-1}\right\|_2^2 \leq \frac{R^2}{2}, \tag{6}$$

$$f\left(y^N\right) - f\left(x_*\right) \leq \frac{R^2}{2A_N}, \quad \left\|u^N - x_*\right\|_2 \leq R, \tag{7}$$

$$\sum_{k=1}^{N} A_k L_{k-1}\left\|y^k - x^{k-1}\right\|_2^2 \leq 2R^2. \tag{8}$$

We also need the following Lemma.

Lemma 3 ((Monteiro and Svaiter, 2013, Lemma 3.7 a)) *Let sequences $\{A_k, L_k\}$, $k \geq 0$ be generated by Algorithm 2. Then, for all $N \geq 0$,*

$$A_N \geq \frac{1}{4}\left(\sum_{k=1}^{N}\frac{1}{\sqrt{L_{k-1}}}\right)^2. \tag{9}$$

Algorithm 2 Accelerated hybrid proximal extragradient method

Input: u_0, y_0 — starting point; N — iteration number; $A_0 = 0$

Output: y^N

1: **for** $k = 0, 1, 2, \ldots, N-1$ **do**

2: Choose L_k and y^{k+1} s.t. $\left\|\nabla F_{L_k,x^k}\left(y^{k+1}\right)\right\|_2 \leq \frac{L_k}{2}\left\|y^{k+1} - x^k\right\|_2$, where

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

3: $u^{k+1} = u^k - a_{k+1}\nabla f\left(y^{k+1}\right)$.

4: **end for**

5: **return** y^N

2.2. Proof of Theorem 1

It follows from Algorithm 1 that $y^{k+1} = T_{p,pM_p}^{F_{L_k,x^k}}(x^k)$, thus by (Nesterov, 2018a, Lemma 1),

$$\left\|\nabla F_{L_k,x^k}\left(y^{k+1}\right)\right\|_2 \leq \frac{(p+1)\,M_p}{p!}\left\|y^{k+1} - x^k\right\|_2^p.$$

At the same time, by the condition in step 2 of Algorithm, 1,

$$\frac{2(p+1)M_p}{p!L_k}\|y^{k+1}-x^k\|_2^{p-1}\leqslant 1.$$

Hence,

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

and we can apply the framework of the previous subsection. What remains is to estimate the growth of A_N, which is our next step.

By the condition in step 2 of Algorithm, 1,

$$\frac{1}{L_k}\left\|y^{k+1}-x^k\right\|_2^{p-1}\geq \theta, \tag{10}$$

where $\theta = p!/(4(p+1)M_p)$. Using this inequality, we prove that

$$\sum_{k=1}^N A_k L_{k-1}^{\frac{p+1}{p-1}} \leq 2R^2\theta^{-\frac{2}{p-1}}. \tag{11}$$

Indeed, from (8) and (10) we have that

$$\begin{aligned}\theta^{\frac{2}{p-1}}\sum_{k=1}^N A_k L_{k-1}^{\frac{p+1}{p-1}} &\leq \sum_{k=1}^N A_k L_{k-1}^{1+\frac{2}{p-1}}\left(\frac{1}{L_{k-1}}\left\|y^k-x^{k-1}\right\|_2^{p-1}\right)^{\frac{2}{p-1}}\\ &= \sum_{k=1}^N A_k L_{k-1}\left\|y^k-x^{k-1}\right\|_2^2\leq 2R^2. \end{aligned}\tag{12}$$

Further, from (11) it follows that

$$\sum_{k=1}^N \frac{1}{\sqrt{L_{k-1}}}\geq \frac{\theta^{\frac{1}{p+1}}}{(2R^2)^{\frac{p-1}{2(p+1)}}}\left(\sum_{k=1}^N A_k^{\frac{p-1}{3p+1}}\right)^{\frac{3p+1}{2(p+1)}}. \tag{13}$$

To prove the relation in (13), let us introduce new variables $z_k = 1\big/\sqrt{L_{k-1}}$ and consider the following optimization problem to find the worst possble value of the l.h.s. in (13)

$$\min\sum_{k=1}^N z_k \quad \text{s.t.} \quad \sum_{k=1}^N A_k z_k^{-\gamma}\leq C, \tag{14}$$

where in accordance to (11)

$$\gamma = 2\frac{p+1}{p-1},\quad C = 2R^2\theta^{-\frac{2}{p-1}}.$$

Since the objective and constraints are separable, this problem can be solved explicitly by the Lagrange principle

$$z_k = \left(\frac{1}{C}\sum_{j=1}^N A_j^{\frac{1}{\gamma+1}}\right)^{1/\gamma} A_k^{\frac{1}{\gamma+1}}.$$

Hence,

$$\min_{\sum_{k=1}^{N} A_k z_k^{-\gamma} \le C} \sum_{k=1}^{N} z_k = \frac{1}{C^{1/\gamma}} \left(\sum_{k=1}^{N} A_k^{\frac{1}{\gamma+1}} \right)^{\frac{\gamma+1}{\gamma}}.$$

From this inequality, (9) and (13), we have

$$A_N \ge \frac{1}{4} \frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}} \left(\sum_{k=1}^{N} A_k^{\frac{p-1}{3p+1}} \right)^{\frac{3p+1}{p+1}}. \tag{15}$$

Moreover, from (15) we obtain that there exists a number c such that, for all $N \ge 0$,

$$A_N \ge \frac{1}{cM_p R^{p-1}} N^{\frac{3p+1}{2}}. \tag{16}$$

The derivation of exact value of the constant c can be found in Lemma 5 in Appendix. This finishes the proof.

2.3. Implementation details

First of all, Theorem 1 in Nesterov (2018a) shows that, by the appropriate choice $M = pM_p$ in (3), the subproblem for finding y^{k+1} in Step 2 of Algorithm 1 is a convex problem and, thus is tractable. Moreover, for $p = 2$ this step corresponds to the step of cubic regularized Newton method of Nesterov and Polyak (2006) and, as it is shown there, can be computed with the same complexity as solving a linear system. For the case $p = 3$, Nesterov (2018a) showed that this step can be also computed efficiently. In both cases the complexity of calculating y^{k+1} is $\tilde{O}\left(n^{2.37}\right)$.

Let us now discuss the process of finding such L_k that the inequality (5) holds. By construction,

$$y^{k+1} = \arg\min_{y \in \mathbb{R}^n} \left\{ \sum_{r=0}^{p} \frac{1}{r!} \nabla^r f\left(x^k\right) \underbrace{\left[y - x^k, ..., y - x^k\right]}_{r} + \frac{pM_p}{(p+1)!} \left\| y - x^k \right\|_2^{p+1} + \frac{L_k}{2} \|y - x^k\|_2^2 \right\}.$$

This problem is strongly convex and, thus, has a unique solution for each $L_k > 0$. Hence, y^{k+1} is uniquely defined by L_k. At the same time, if $L_k \to 0$, $y^{k+1} \to \tilde{y}^k$ with

$$\tilde{y}^k \in \operatorname{Arg}\min_{y \in \mathbb{R}^n} \left\{ \sum_{r=0}^{p} \frac{1}{r!} \nabla^r f\left(x^k\right) \underbrace{\left[y - x^k, ..., y - x^k\right]}_{r} + \frac{pM_p}{(p+1)!} \left\| y - x^k \right\|_2^{p+1} \right\}$$

being a fixed point. Whence,

$$\frac{2(p+1)M_p}{p!L_k} \|y^{k+1} - x^k\|_2^{p-1} \to +\infty.$$

On the other hand, if $L_k \to +\infty$, $y^{k+1} \to x^k$ and

$$\frac{2(p+1)M_p}{p!L_k} \|y^{k+1} - x^k\|_2^{p-1} \to 0.$$

By the continuity of the dependence of y^{k+1} from L_k, we see that there exists such L_k that inequality (5) holds. An appropriate value of L_k can be found by an extended line-search procedure as in (Monteiro and Svaiter, 2013, Section 7). The details of complexity of the line-search can be found in (Jiang et al., 2018; Bubeck et al., 2018), where the authors prove a bound of $\tilde{O}(1)$ calls of $T_{p,pM_p}^{F_{L_k,x^k}}(x^k)$ on each iteration.

3. Extension for Uniformly Convex Case

In this section, we additionally assume that the objective function is uniformly convex of degree $q \geq 2$ on the convex bounded set Q, i.e., there exists $\sigma_q > 0$ s.t.

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\sigma_q}{q} \|y - x\|_2^q, \quad \forall x, y \in Q. \tag{17}$$

Note that for the case $q = 2$ we get the definition of σ_2-strongly convex function. As we mentioned before in the Section 1 we can take $Q = B_{R_1}(x^*)$ with $R_1 = O(R)$. In this case we will get that exist non-trivial functions satisfying both uniform convexity with $q > 2$ and Lipschitz continuity of p-th derivatives for $p < q - 1$. One can check that for $q > 2$ function $\frac{1}{q}\|x - x_0\|^q$ is a uniformly convex function of degree $q > 2$.

We also assume that $q \leq p + 1$. As a corollary,

$$f(y) \geq f(x_*) + \frac{\sigma_q}{q} \|y - x_*\|_2^q, \forall y \in Q, \tag{18}$$

where x_* is a solution to problem (1). We show how the restart technique can be used to accelerate Algorithm 1 under this additional assumption.

Algorithm 3 Restarted Optimal Tensor Method

Input: p, M_p, q, σ_q, z_0, Δ_0 s.t. $f(z^0) - f^* \leq \Delta_0$.

1: **for** $k = 0, 1, ...$ **do**

2:

$$\text{Set} \quad \Delta_k = \Delta_0 \cdot 2^{-k} \quad \text{and} \quad N_k = \max\left\{ \left\lceil \left(\frac{2cM_p q^{\frac{p+1}{q}}}{\sigma_q^{\frac{p+1}{q}}} \Delta_k^{\frac{p+1-q}{q}} \right)^{\frac{2}{3p+1}} \right\rceil, 1 \right\}. \tag{19}$$

3: Set $z_{k+1} = y^{N_k}$ as the output of Algorithm 1 started from z_k and run for N_k steps.

4: Set $k = k + 1$.

5: **end for**

Output: z_k.

Theorem 4 *Let sequence z^k, $k \geq 0$ be generated by Algorithm 3. Then*

$$\frac{\sigma_q}{q} \|z_k - x_*\|_2^q \leq f(z_k) - f^* \leq \Delta_0 \cdot 2^{-k},$$

and the total number of steps of Algorithm 1 is bounded by (c is defined in (16))

$$\left(2cq^{\frac{p+1}{q}}\right)^{\frac{2}{3p+1}} \frac{M_p^{\frac{2}{3p+1}}}{\sigma_q^{\frac{2(p+1)}{q(3p+1)}}} (\Delta_0)^{\frac{2(p+1-q)}{q(3p+1)}} \cdot \sum_{i=0}^{k} 2^{-i\frac{2(p+1-q)}{q(3p+1)}} + k.$$

Proof Let us prove the first statement of the Theorem by induction. For $k = 0$ it holds. If it holds for some $k \geq 0$, by the choice of N_k, we have that

$$\frac{cM_p}{N_k^{\frac{3p+1}{2}}} \left(\frac{q\Delta_k}{\sigma_q}\right)^{\frac{p+1}{q}} \leq \frac{\Delta_k}{2}.$$

By (18),

$$\|z_k - x_*\|_2^{p+1} \leq \left(\frac{q(f(z_k) - f^*)}{\sigma_q}\right)^{\frac{p+1}{q}} \leq \left(\frac{q\Delta_k}{\sigma_q}\right)^{\frac{p+1}{q}}$$

since, by our assumption, $q \leq p + 1$. Combining the above two inequalities and Theorem 1, we obtain

$$f(z_{k+1}) - f^* \leq \frac{cM_p\|z_k - x_*\|_2^{p+1}}{N_k^{\frac{3p+1}{2}}} \leq \frac{\Delta_k}{2} = \Delta_{k+1}.$$

It remains to bound the total number of steps of Algorithm 1. Denote $\tilde{c} = \left(2cq^{\frac{p+1}{q}}\right)^{\frac{2}{3p+1}}$.

$$\sum_{i=0}^{k} N_i \leq \tilde{c}\frac{M_p^{\frac{2}{3p+1}}}{\sigma_q^{\frac{2(p+1)}{q(3p+1)}}} \sum_{i=0}^{k} (\Delta_0 \cdot 2^{-i})^{\frac{2(p+1-q)}{q(3p+1)}} + k \leq \tilde{c}\frac{M_p^{\frac{2}{3p+1}}}{\sigma_q^{\frac{2(p+1)}{q(3p+1)}}} (\Delta_0)^{\frac{2(p+1-q)}{q(3p+1)}} \cdot \sum_{i=0}^{k} 2^{-i\frac{2(p+1-q)}{q(3p+1)}} + k.$$

■

Let us make several remarks on the complexity of the restarted scheme in different settings. It is easy to see from Theorem 4 that, to achieve an accuracy ε, i.e., to find a point $\hat{x}$ s.t. $f(\hat{x}) - f^* \leq \varepsilon$, the number of tensor steps in Algorithm 3 is

$$O\left(\frac{M_p^{\frac{2}{3p+1}}}{\sigma_q^{\frac{2(p+1)}{q(3p+1)}}} (\Delta_0)^{\frac{2(p+1-q)}{q(3p+1)}} + \log_2 \frac{\Delta_0}{\varepsilon}\right), q < p+1, \text{and } O\left(\left(\frac{M_p^{\frac{2}{3p+1}}}{\sigma_q^{\frac{2(p+1)}{q(3p+1)}}} + 1\right) \log_2 \frac{\Delta_0}{\varepsilon}\right), q = p+1.$$

Theorem 4 suggests a natural generalization of first- and second-order condition number Nesterov (2008). If f is such that $q = p + 1$, then the complexity of Algorithm 3 depends only logarithmically on the starting point and is proportional to $(\gamma_p)^{\frac{2}{3p+1}}$, where $\gamma_p = M_p/\sigma_{p+1}$ is the p-th order condition number. Unfortunately, if $q < p + 1$, the complexity depends polynomially on the initial objective residual Δ_0, which, in general, is not controlled.

An interesting special case is when $q = 2$ and $p \geq 2$, and, as a consequence, $q < p + 1$. As it can be seen from Theorem 2 (see also Bubeck et al. (2018)), the sequence, generated by Algorithm 1 is bounded by some $R = O(\|x^0 - x_*\|_2)$. Hence, the constant M_2 can be estimated as $M_2 \leq M_p R^{p-2}$. At the same time, in (Nesterov, 2008, Sect.6), it is shown that the Cubic regularized Newton method Nesterov and Polyak (2006) has the region of quadratic convergence given by $\{x : f(x) - f^* \le \sigma_2^2/(2M_2^2) \le \sigma_2^2/(2M_p^2 R^{2(p-2)})\}$. To enter this region, Algorithm 3 requires

$$O\left(\frac{M_p^{\frac{2}{3p+1}}}{\sigma_2^{\frac{p+1}{3p+1}}}(\Delta_0)^{\frac{p-1}{3p+1}} + \log_2 \frac{\Delta_0 M_p^2 R^{2(p-2)}}{\sigma_2^2}\right) = O\left(\frac{M_p^{\frac{2}{3p+1}}}{\sigma_2^{\frac{p+1}{3p+1}}}(\Delta_0)^{\frac{p-1}{3p+1}} + \log_2 \frac{M_p^2 \Delta_0^{p-1}}{\sigma_2^p}\right), \tag{20}$$

where we used inequality $R^2 \le 2\Delta_0/\sigma_2$, which follows from (18). After entering the region of quadratic convergence, Algorithm 3 can be switched to the Cubic regularized Newton method Nesterov and Polyak (2006), which has final stage complexity, (Nesterov and Polyak, 2006, Sect. 6)

$$O\left(\log_{3/2}\log_4 \frac{\sigma_2^3}{M_2^2\varepsilon}\right) = O\left(\log_{3/2}\log_4 \frac{\sigma_2^3}{M_p^2 R^{2(p-2)}\varepsilon}\right).$$

Summing this inequality and (20) we obtain the total complexity of this switching procedure to obtain small accuracy ε. Note, that the second term in (20) is typically dominated by the first one, so we can ignore it without loss of generality.

Finally, let us compare our upper bound with known lower bounds. For the case $p = 1$, $q = 2$, our complexity bound coincides with lower bound for first-order methods Nemirovsky and Yudin (1983); Nesterov (2004). Arjevani et al. (2018) propose lower bounds for second-order methods for the case $p = 2$, $q = 2$ and our complexity bound coincides with their lower bound up to a change of $D = \sqrt{\Delta_0/\sigma_2}$, which is natural as, in this case f is strongly convex. For the cases of $p \notin \{1, 2\}, q \neq 2$ we are not aware of any lower bounds.

4. Numerical Analysis

In this section, we analyze and compare the performance of Algorithm 1 with the accelerated tensor method proposed in Nesterov (2018a). Each of the subproblems, where y_k is computed, is solved using the implementation details as in Nesterov (2018a), where the auxiliary problem is reduced to a scalar optimization problem that can be solved fast using existing computational tools. Also, at each iteration, the value of L_k is selected according to a simple line search as described in (Bubeck et al., 2018).

We study the numerical performance for two classes of functions. Initially, an universal parametric family of objective functions, which are difficult for all tensor methods Nesterov (2018a) defined as

$$f_m(x) = \eta_{p+1}(A_m x) - x_1, \tag{21}$$

where, for integer parameter $p \ge 1$, $\eta_{p+1}(x) = \frac{1}{p+1}\sum_{i=1}^{n} |x_i|^{p+1}$, $2 \le m \le n$, $x \in \mathbb{R}^n$, A_m is the $n \times n$ block diagonal matrix:

$$A_m = \begin{pmatrix} U_m & 0 \\ 0 & I_{n-m} \end{pmatrix}, \quad \text{with} \quad U_m = \begin{pmatrix} 1 & -1 & 0 & \dots & 0 \\ 0 & 1 & -1 & \dots & 0 \\ \vdots & \vdots & \ddots & & \vdots \\ 0 & 0 & \dots & 1 & -1 \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix}, \tag{22}$$

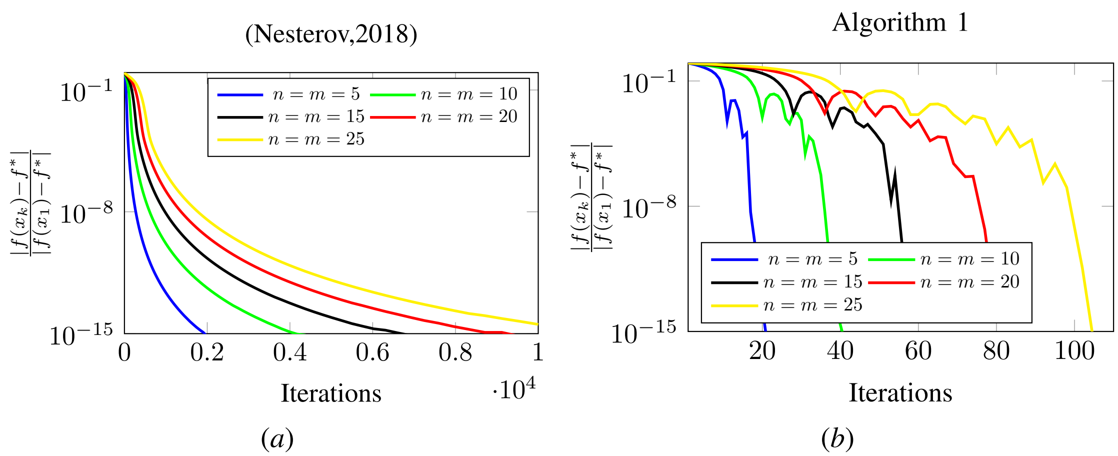

Figure 1: A performance comparison between the accelerated tensor method in Nesterov (2018a) (shown in (a)) and Algorithm 1 (shown in (b)). We minimize an instance of the family of functions in (21) with $p = 3$ and various values of dimension n and k. Note that the x-axis scaling on both figures is different.

and I_n is the identity $n \times n$-matrix. For a detailed description of the high-order derivatives of this class of functions, and its optimality properties see Nesterov (2018a).

Figure 1 shows the normalized optimality gap of the iterations generated by the accelerated tensor method from Nesterov (2018a) in Figure1(a), and Algorithm 1 in Figure1(b). We denote the minimum function value as f^*. For both results we have used $p = 3$, and $n = m = \{5, 10, 15, 20, 25\}$. These numerical results show that Algorithm 1 requires a much smaller number of iterations than the accelerated tensor method from Nesterov (2018a) to reach the same optimality gap, namely $1 \cdot 10^{-15}$, for the class of "bad" functions described in Nesterov (2018a). For example, for the case where $n = m = 25$, Algorithm 1 has reached the desired accuracy in about 100 iterations, while the accelerated tensor method requires about $1 \cdot 10^4$.

As a second set of numerical results we study the performance of the proposed method for the non-regularized logistic regression problem, again with $p = 3$. For this problem we are given a set of d data pairs $\{y_i, w_i\}$ for $1 \leq i \leq d$, where $y_i \in \{1, -1\}$ is the class label of object i, and $w_i \in \mathbb{R}^n$ is the set of features of object i. We are interested in finding a vector x that solves the following optimization problem

$$\frac{1}{d}\sum_{i=1}^{d} \ln\Big(1 + \exp\big(-y_i\langle w_i, x\rangle\big)\Big) \to \min_{x\in\mathbb{R}^n}. \tag{23}$$

Figure 2 shows the simulation results for the logistic regression problem in (23) for various datasets. Similarly as in Figure 1, we compare the performance of Algorithm 1, and the accelerated tensor method in Nesterov (2018a). In Figure 2(a) and Figure 2(b), we generate synthetic data, where, initially we define a vector $\hat{x} \in [-1, 1]$ with every entry is chosen uniformly at random. The set of features for each i, i.e., $w_i \in [-1, 1]^n$ has also every entry chosen uniformly at random, finally each label is computed as $y_i = \text{sign}(\langle w_i, \hat{x}\rangle)$. For Figure 2(a) we set $n = 10$ and $d = 100$, while in Figure 2(b) we set $n = 100$ and $d = 1000$. Figure 2(c) uses the `mushroom` dataset ($n = 8124$ and

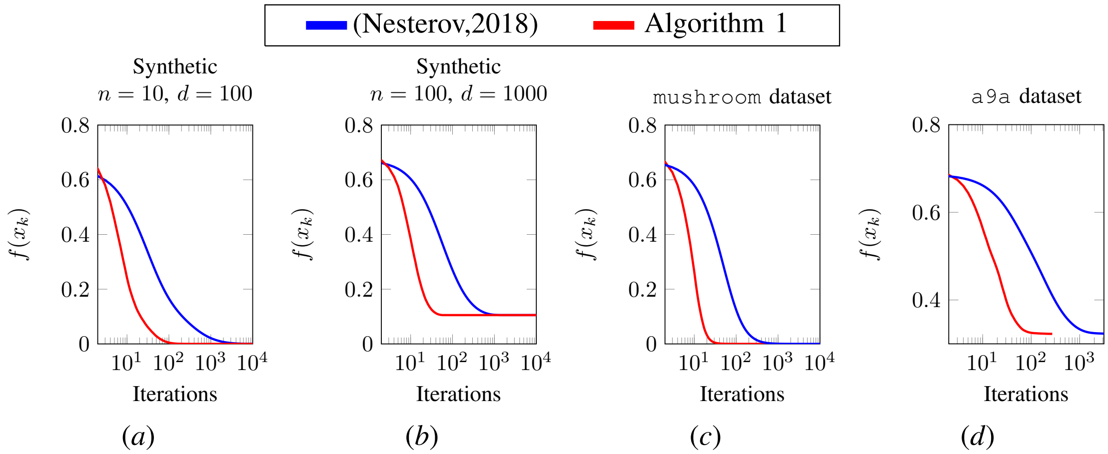

Figure 2: A performance comparison for the non-regularized logistic regression problem between the accelerated tensor method from Nesterov (2018a) and Algorithm 1. (a) Uses synthetic data with $n = 10$ and $d = 100$, (b) uses synthetic data with $n = 100$ and $d = 1000$, (c) uses the `mushroom` dataset ($d = 8124$ and $n = 112$) Dheeru and Karra Taniskidou (2017), and (d) uses the `a9a` dataset ($d = 32561$ and $n = 123$) Dheeru and Karra Taniskidou (2017).

$d = 112$) Dheeru and Karra Taniskidou (2017), and Figure 2(d) uses the `a9a` dataset ($n = 32561$ and $d = 123$) Dheeru and Karra Taniskidou (2017).

For the logistic regression problem, we don't have access to the optimal value function in general, thus, we plot only the cost function evaluated at the current iterate. As expected by the theoretic results, Algorithm 1 requires one order of magnitude less iterations than the accelerated tensor method from Nesterov (2018a) to achieve the same function value.

In Appendix B, we numerically compare the performance of the accelerated tensor method from Nesterov (2018a) for $p = 2$ and $p = 3$, as well as its accelerated and non-accelerated versions.

Acknowledgments

The authors are grateful to Yurii Nesterov for fruitful discussions. We notice that Yurii Nesterov in his book Nesterov (2018b) (see Section 4.3) provides very similar scheme to ours but with cubic Newton step instead of tensor step in Monteiro-Svaiter approach.

The work of A. Gasnikov in Section 2 was supported by RFBR 18-31-20005 mol_a_ved, the work in Section 3 was funded by Russian Science Foundation (project 17-11-01027), the work of E. Vorontsova in section 4 was supported by 18-29-03071 mk, the work of E. Gorbunov in Appendix A was supported by grant of Russian president MD 1320.2018.1.

References

Naman Agarwal and Elad Hazan. Lower bounds for higher-order convex optimization. In Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet, editors, *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, pages 774–792. PMLR, 06–09 Jul 2018. URL http://proceedings.mlr.press/v75/agarwal18a.html.

Yossi Arjevani, Ohad Shamir, and Ron Shiff. Oracle complexity of second-order methods for smooth convex optimization. *Mathematical Programming*, May 2018. ISSN 1436-4646. doi: 10.1007/s10107-018-1293-1. URL https://doi.org/10.1007/s10107-018-1293-1.

Michel Baes. Estimate sequence methods:extensions and approximations. Technical report, 2009. URL http://www.optimization-online.org/DB_FILE/2009/08/2372.pdf.

E. G. Birgin, J. L. Gardenghi, J. M. Martínez, S. A. Santos, and Ph. L. Toint. Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models. *Mathematical Programming*, 163(1):359–368, May 2017. ISSN 1436-4646. doi: 10.1007/s10107-016-1065-8. URL https://doi.org/10.1007/s10107-016-1065-8.

Sébastien Bubeck, Qijia Jiang, Yin Tat Lee, Yuanzhi Li, and Aaron Sidford. Near-optimal method for highly smooth convex optimization. *arXiv:1812.08026*, 2018.

Coralia Cartis, Nicholas I. M. Gould, and Philippe L. Toint. Improved second-order evaluation complexity for unconstrained nonlinear optimization using high-order regularized models. *arXiv:1708.04044*, 2018.

Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Nikita Doikov and Yurii Nesterov. Minimizing uniformly convex functions by cubic regularization of newton method. *arXiv preprint arXiv:1905.02671*, 2019.

Geovani Nunes Grapiglia and Yurii Nesterov. Tensor methods for minimizing functions with h\"{o} lder continuous higher-order derivatives. *arXiv preprint arXiv:1904.12559*, 2019.

K. H. Hoffmann and H. J. Kornstaedt. Higher-order necessary conditions in abstract mathematical programming. *Journal of Optimization Theory and Applications*, 26(4):533–568, Dec 1978. ISSN 1573-2878. doi: 10.1007/BF00933151. URL https://doi.org/10.1007/BF00933151.

Bo Jiang, Haoyue Wang, and Shuzhong Zhang. An optimal high-order tensor method for convex optimization. *arXiv:1812.06557*, 2018.

R. Monteiro and B. Svaiter. An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. *SIAM Journal on Optimization*, 23(2):1092–1125, 2013. doi: 10.1137/110833786. URL https://doi.org/10.1137/110833786.

A.S. Nemirovsky and D.B. Yudin. *Problem Complexity and Method Efficiency in Optimization*. J. Wiley & Sons, New York, 1983.

Yu. Nesterov. Accelerating the cubic regularization of newton's method on convex problems. *Mathematical Programming*, 112(1):159–181, Mar 2008. ISSN 1436-4646. doi: 10.1007/s10107-006-0089-x. URL https://doi.org/10.1007/s10107-006-0089-x.

Yurii Nesterov. A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Soviet Mathematics Doklady*, 27(2):372–376, 1983.

Yurii Nesterov. *Introductory Lectures on Convex Optimization: a basic course*. Kluwer Academic Publishers, Massachusetts, 2004.

Yurii Nesterov. Implementable tensor methods in unconstrained convex optimization. Technical report, CORE UCL, 2018a. URL https://alfresco.uclouvain.be/alfresco/service/guest/streamDownload/workspace/SpacesStore/aabc2323-0bc1-40d4-9653-1c29971e7bd8/coredp2018_05web.pdf. CORE Discussion Paper 2018/05.

Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018b.

Yurii Nesterov and Boris Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006. ISSN 1436-4646. doi: 10.1007/s10107-006-0706-8. URL http://dx.doi.org/10.1007/s10107-006-0706-8.

Andre Wibisono, Ashia C. Wilson, and Michael I. Jordan. A variational perspective on accelerated methods in optimization. *Proceedings of the National Academy of Sciences*, 113(47):E7351–E7358, 2016.

Ashia Wilson, Lester Mackey, and Andre Wibisono. Accelerating rescaled gradient descent. *arXiv preprint arXiv:1902.08825*, 2019.

Optimal Tensor Methods in Smooth Convex and Uniformly Convex Optimization: Supplementary Material

Appendix A. Technical lemmas

Lemma 5 *Consider the sequence $\{A_k\}_{k\geq 0}$ of non-negative numbers such that*

$$A_N \geq \frac{1}{4}\frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}}\left(\sum_{k=1}^{N} A_k^{\frac{p-1}{3p+1}}\right)^{\frac{3p+1}{p+1}}, \tag{24}$$

where $p \geq 3$, $\theta = p!/(4(p+1)M_p)$ and $M_p, R > 0$. Then for all $N \geq 0$, we have

$$A_k \geq \frac{1}{cM_pR^{p-1}} k^{\frac{3p+1}{2}}, \tag{25}$$

where

$$c = \frac{2^{\frac{3(p+1)^2+4}{4}}(p+1)}{p!}. \tag{26}$$

Proof We prove (25) by induction. For $k = 1$ we have

$$A_1 \overset{(24)}{\geq} \frac{1}{4}\frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}}A_1^{\frac{p-1}{p+1}} \iff A_1^{\frac{2}{p+1}} \geq \frac{1}{4}\frac{\theta^{\frac{2}{p+1}}}{2^{\frac{p-1}{p+1}}R^{\frac{2(p-1)}{p+1}}} \iff A_1 \geq \frac{p!}{2^{\frac{3p+5}{2}}(p+1)M_pR^{p-1}}.$$

The last inequality implies (25) for $p \geq 3$. Now let us assume that for all $k \leq N$ inequality (25) holds and $N \geq 1$. Next we will establish (25) for $k = N + 1$. We have

$$\begin{aligned} A_{N+1} &\overset{(24)}{\geq} \frac{1}{4}\frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}}\left(\sum_{k=1}^{N+1} A_k^{\frac{p-1}{3p+1}}\right)^{\frac{3p+1}{p+1}} \\ &\geq \frac{1}{4}\frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}}\left(\sum_{k=1}^{N} A_k^{\frac{p-1}{3p+1}}\right)^{\frac{3p+1}{p+1}} \\ &\overset{(25)}{\geq} \frac{1}{4}\frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}}\left(\left(\frac{1}{cM_pR^{p-1}}\right)^{\frac{p-1}{3p+1}}\sum_{k=1}^{N} k^{\frac{p-1}{2}}\right)^{\frac{3p+1}{p+1}}. \end{aligned}$$

If $N = 1$, then

$$A_{N+1} = A_2 \geq \frac{1}{2^{\frac{3p+1}{2}}}\frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}}\left(\frac{1}{cM_pR^{p-1}}\right)^{\frac{p-1}{p+1}}(2)^{\frac{3p+1}{2}}. \tag{27}$$

If $N > 1$, we can write

$$A_{N+1} \geq \frac{1}{4}\frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}}\left(\frac{1}{cM_pR^{p-1}}\right)^{\frac{p-1}{p+1}}\left(1+\sum_{k=2}^{N} k^{\frac{p-1}{2}}\right)^{\frac{3p+1}{p+1}}. \tag{28}$$

Since $(p-1)/2 \geq 1$ the function $f(x) = x$ is convex and, as a consequence, we get

$$\sum_{k=2}^{N} k^{\frac{p-1}{2}} \geq \int_{1}^{N} x^{\frac{p-1}{2}} dx = \frac{2}{p+1} N^{\frac{p+1}{2}} - \frac{2}{p+1} \geq \frac{2}{p+1} N^{\frac{p+1}{2}} - \frac{1}{2}. \tag{29}$$

Using this fact we continue:

$$\begin{aligned} A_{N+1} &\overset{(29)}{\geq} \frac{1}{4} \frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}} \left(\frac{1}{cM_p R^{p-1}}\right)^{\frac{p-1}{p+1}} \left(\frac{1}{2} + N^{\frac{p+1}{2}}\right)^{\frac{3p+1}{p+1}} \\ &\geq \frac{1}{4} \frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}} \left(\frac{1}{cM_p R^{p-1}}\right)^{\frac{p-1}{p+1}} N^{\frac{3p+1}{2}}. \end{aligned}$$

For all $N > 1$ we have

$$\left(\frac{N}{N+1}\right)^{\frac{3p+1}{2}} = \left(1 - \frac{1}{N+1}\right)^{\frac{3p+1}{2}} \geq \left(1 - \frac{1}{2}\right)^{\frac{3p+1}{2}} = \frac{1}{2^{\frac{3p+1}{2}}}.$$

From this and (28) we obtain that for all $N \geq 1$

$$A_{N+1} \geq \frac{1}{2^{\frac{3p+1}{2}}} \frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}} \left(\frac{1}{cM_p R^{p-1}}\right)^{\frac{p-1}{p+1}} (N+1)^{\frac{3p+1}{2}}.$$

It remains to show that (26) implies

$$\frac{1}{2^{\frac{3p+1}{2}}} \frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}} \left(\frac{1}{cM_p R^{p-1}}\right)^{\frac{p-1}{p+1}} = \frac{1}{cM_p R^{p-1}}.$$

Using $\theta = p!/(4(p+1)M_p)$ we get

$$\begin{aligned} \frac{1}{2^{\frac{3p+1}{2}}} \frac{\theta^{\frac{2}{p+1}}}{(2R^2)^{\frac{p-1}{p+1}}} \left(\frac{1}{cM_p R^{p-1}}\right)^{\frac{p-1}{p+1}} = \frac{1}{cM_p R^{p-1}} &\iff c^{\frac{2}{p+1}} \frac{1}{2^{\frac{3p+1}{2}}} \left(\frac{p!}{4(p+1)}\right)^{\frac{2}{p+1}} \frac{1}{2^{\frac{p-1}{p+1}}} = 1 \\ \iff c^{\frac{2}{p+1}} = 2^{\frac{3p+1}{2}} \left(\frac{4(p+1)}{p!}\right)^{\frac{2}{p+1}} 2^{\frac{p-1}{p+1}} &\iff c = 2^{\frac{(3p+1)(p+1)}{4}} \frac{4(p+1)}{p!} 2^{\frac{p-1}{2}} \\ &\iff c = \frac{2^{\frac{3(p+1)^2+4}{4}}(p+1)}{p!}, \end{aligned}$$

which is exactly what we have in (26). ■

Appendix B. Comparison of the accelerated tensor method from Nesterov (2018a) for $p = 2$ and $p = 3$.

In this appendix, we numerically compare the performance of the accelerated tensor method proposed in (Nesterov, 2018a), for $p = 2$ and $p = 3$. We also compare the accelerated and non-accelerated version of this method.

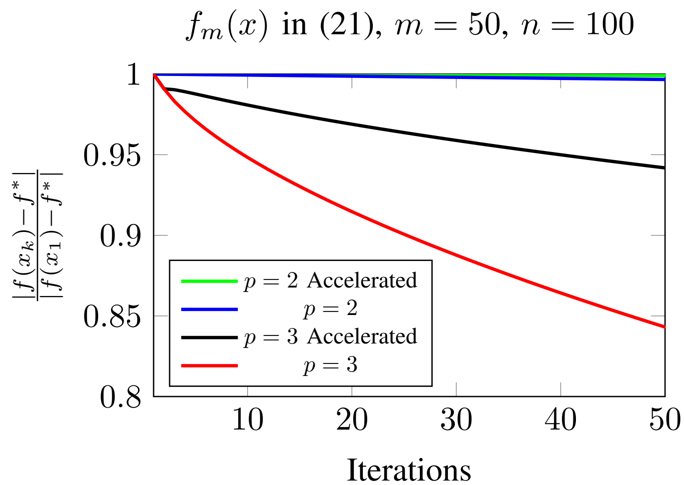

Figure 3: Performance of tensor methods and accelerated tensor methods for $p = 2$ and $p = 3$ on a difficult instance (21) for all unconstrained minimization tensor methods with $n = 100$ and $m = 50$.

Similarly as in Figure 1 and Figure 2, we present the numerical results for the class of bad functions defined in (21) and one instance of the logistic regression problem.

In Figure 3, we compare the behavior of the following methods: 1) tensor method Nesterov (2018a) for $p = 3$; 2) accelerated tensor method Nesterov (2018a) for $p = 3$; 3) tensor method Nesterov (2018a) for $p = 2$; 4) accelerated tensor method Nesterov (2018a) for $p = 2$. Again, the optimal function value is denoted by f^*. Interestingly, we obtain that the non-accelerated method outperforms the accelerated method for the first m iterations. Since Theorem 4 from Nesterov (2018a) works only for $k \leq m$ we don't study the behaviour of the methods for larger number of iterations. Even in this simple setting it is still non-trivial how to implement tensor methods for such bad examples of functions.

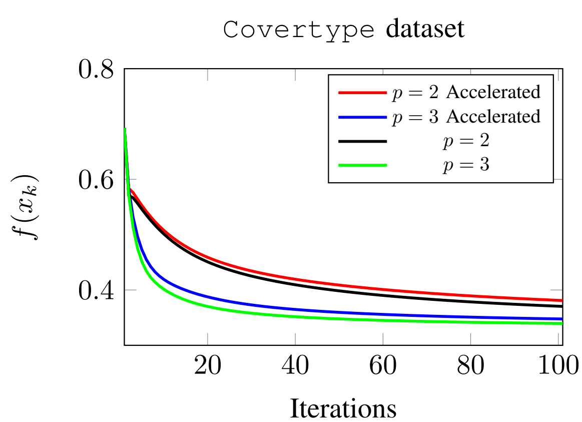

Figure 4: Function value achieved by the iterates of the accelerated tensor method for the logistic regression problem on the `Covertype` dataset Dheeru and Karra Taniskidou (2017). Number of samples $d = 20000$, dimension $n = 55$.

In Figure 4, we consider the behaviour of the same set of methods as in Figure 3, but for logistic regression problem defined in (23) on Covertype dataset Dheeru and Karra Taniskidou (2017). And again, we notice that in both cases non-accelerated version works better in our experiments

First of all, we point out that tensor methods in general are non-trivial in implementation, so, it is interesting direction of the future work to get better implementation. Secondly, we conjecture that slow convergence that we see in our experiments is because of large M_p that we use. Due to tuning of the parameters one can obtain better convergence in practice.