8-28-2022

Exploiting higher-order derivatives in convex optimization methods

Dmitry Kamzolov
Mohamed bin Zayed University of Artificial Intelligence

Alexander Gasnikov
MIPT, Moscow, Russian Federation & IITP RAS, Moscow, Russian Federation & Caucasus Mathematical Center, Adyghe State University, Maikop, Russian Federation

Pavel Dvurechensky
WIAS, Berlin, Germany

Artem Agafonov
Mohamed Bin Zayed University of Artificial Intelligence, Abu Dhabi, United Arab Emirates & MIPT, Moscow, Russian Federation

Martin Takac
Mohamed bin Zayed University of Artificial Intelligence

Exploiting higher-order derivatives in convex optimization methods

Dmitry Kamzolov (Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE), Alexander Gasnikov (MIPT, Moscow, Russia; IITP RAS, Moscow, Russia; Caucasus Mathematical Center, Adyghe State University, Maikop, Russia), Pavel Dvurechensky (WIAS, Berlin, Germany), Artem Agafonov (Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE; MIPT, Moscow, Russia), Martin Takáč (Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE)

Exploiting higher-order derivatives in convex optimization is known at least since 1970's. In each iteration higher-order (also called tensor) methods minimize a regularized Taylor expansion of the objective function, which leads to faster convergence rates if the corresponding higher-order derivative is Lipschitz-continuous. Recently a series of lower iteration complexity bounds for such methods were proved, and a gap between upper an lower complexity bounds was revealed. Moreover, it was shown that such methods can be implementable since the appropriately regularized Taylor expansion of a convex function is also convex and, thus, can be minimized in polynomial time. Only very recently an algorithm with optimal convergence rate $1/k^{(3p+1)/2)}$ *was proposed for minimizing convex functions with Lipschitz p-th derivative. For convex functions with Lipschitz third derivative, these developments allowed to propose a second-order method with convergence rate* $1/k^5$*, which is faster than the rate* $1/k^{3.5}$ *of existing second-order methods.*

1 Introduction

It is well known since the works of I. Newton [68] and L. Kantorovich [45] that the second-order derivative of the objective function can be used in numerical algorithms for solving optimization problems and nonlinear equations and that such algorithms have better convergence guarantees. Higher-order derivatives can be efficiently used for solving nonlinear equations as was shown by P. Chebyshev [17, 25]. For optimization problems, the basic idea known at least since 1970's [39] and developed in later works [75, 74, 9] is to approximate, at each iteration of an algorithm, the objective by its Taylor polynomial at the current iterate, optionally add a regularization, and minimize this Taylor approximation to obtain the next iterate. In this way, the first Taylor polynomial leads to first-order methods that are very well understood, see, e.g., [58, 62], with the optimal methods existing since 1980's [58, 60]. If the second Taylor polynomial is used, we are in the world of second-order methods with the most famous representative being the *Newton's method* that minimizes at each iteration the second-order quadratic approximation of the objective function. If the second derivative is Lipschitz continuous, the objective is strongly convex and the starting point is sufficiently close to the solution, this algorithm has very fast quadratic convergence and requires $\log\log\varepsilon^{-1}$ iterations to reach an ε-solution in terms of the objective value [45]. This bound is optimal [58] even for univariate optimization problems with the possibility of using in algorithms derivatives of any order. Different modifications of the basic algorithm, such as the Damped Newton's method or the Levenberg–Marquardt algorithm achieve global convergence, but have a slower, i.e., linear, convergence [71, 69]. Second-order methods played also the central role in the development by Yu. Nesterov and A. Nemirovskii of interior point methods that have global linear convergence rate and allow proving polinomial solvability of a large class of convex optimization problems [66]. This theory was based on the analysis of the Damped Newton's method for the class of self-concordant functions that, in particular, include functions without Lipschitz derivatives.

An important idea that eventually led to the current developments of tensor methods was the cubic regularization of the Newton's method, which dates back to the work of A. Griewank [37]. The global performance of the Cubic regularized Newton's method was analysed in 2006 by Yu. Nesterov and B. Polyak in [67] for a large list of settings with the main assumption that the second derivative is Lipschitz continuous. The main idea is based on the fact that, for functions with Lipshitz second derivative, the objective's model consisting of the second Taylor polynomial regularized by the cube of the norm with sufficiently large regularization parameter is an upper bound for the objective function. This model is minimized in each iteration of the algorithm, which, in particular, allowed to obtain global convergence rate $1/k^2$ for minimizing convex functions with Lipschitz second derivative. Moreover, the authors showed that the complexity of minimizing the third-order polynomial model of the objective is of the same order as the complexity of the standard Newton's method step. The Cubic regularized Newton's method was further accelerated in [59] to reach $1/k^3$ convergence rate for convex problems with Lipschitz second derivative. In 2012 R. Monteiro and B. Svaiter proposed in [56] a very perspective *accelerated-proximal envelope* (see also the work [16]) that allowed them to develop even faster second-order method with the convergence rate $1/k^{3.5}$ for minimizing convex objectives with Lipschitz second derivative.

Another important step in the development of tensor methods was made by M. Baes in 2009 [7], where he generalized the Newton–Nesterov–Polyak algorithm to the setting of convex minimization with the objective having Lipschitz p-th derivative for $p \geq 2$. The idea was to construct a $(p+1)$-th order polynomial model that upper bounds the objective function by taking the p-th Taylor polynomial of the objective and regularizing it by the $(p+1)$-th power of the Euclidean norm with sufficiently large regularization parameter. The author showed $1/k^p$ global convergence rate for methods that minimize such model in each iteration and proposed an accelerated version with the rate $1/k^{p+1}$, all under the assumption of Lipschitz p-th derivative. As in the world of first-order methods, where the interest in optimal methods was one of the central driving forces in 2000–2020, a natural question arose on what are the lower bounds for second- and higher-order meth-

ods and which algorithms are optimal. The results in [6, 4, 30] gave the answer that the lower bound is $1/k^{(3p+1)/2}$, revealing the gap when $p \geq 3$ between the rates of existing methods an lower bounds. One of the drawbacks of tensor methods at this stage was that each iteration required to minimize a higher-order polynomial that may not be convex, leading to the high cost of each iteration and impracticality of such methods.

In 2018, Yu. Nesterov [63] made a breakthrough in the understanding of the role of higher-order methods in modern convex optimization when p-th derivative is Lipschitz. He increased the regularization parameter in the regularized p-th Taylor polynomial of the objective in M. Baes's approach and showed that this new Taylor model is convex. Moreover, he showed that for $p = 3$ this convex Taylor model can be minimized very efficiently with almost the same complexity as the standard Newton's method step. Further, he demonstrated that minimzation of the convex Taylor model does not require calculating the full tensor of third derivatives. Instead, it is sufficient to calculate directional third derivatives, e.g., by automatic differentiation. Finally, he proved the lower bound $1/k^{(3p+1)/2}$ for convex problems with Lipschitz p-th derivative, constructed for each $p \geq 2$ the worst-case functions in the class of convex functions with Lipschitz p-th derivatives, and described a research program to develop optimal tensor methods [62]. It followed from the obtained lower bounds for convex optimization problems that the Monteiro–Svaiter second-order method is optimal up to a logarithmic factor since it used a complicated line-search procedure. Based on the Monteiro–Svaiter method and the convex Taylor model of the objective, in [31, 27, 28] and independently in [11, 42], the authors proposed near-optimal tensor methods for convex problems with Lipschitz p-th derivatives, which have the rate $1/k^{(3p+1)/2}$ up to logarithmic factors. We describe the Monteiro–Svaiter approach with Nesterov's implementable tensor steps in Section 3. As it was mentioned in [62, 21], the auxiliary line-search procedures that lead to logarithmic factors in the convergence rates of the near-optimal tensor methods may also significantly slow down the convergence in practice. In 2022, two independent works [48] and [14] proposed optimal tensor methods with convergence rates without additional logarithmic factors. At the same time, such methods were proposed also for monotone variational inequalities [1, 53] under higher-order regularity of the operator.

The developments in the theory of tensor methods had also an important impact on the theory of second-order methods. In 2020, Yu. Nesterov proposed an implementation of the third-order method using inexact third derivative constructed via finite differences of gradients [65]. In other words, it appeared to be possible to implement the third-order method using only first and second derivatives, which lead to «superfast» second-order method with the convergence rate $1/k^4$ violating the lower bound thanks to additional assumption that the third derivative is Lipschitz continuous. The key observation was that in the class of functions with Lipschitz second derivative the worst-case function [63] for second-order methods does not have Lipschitz third derivative. These results were further improved in [43, 64], where second-order algorithms with the rate $1/k^5$ corresponding to optimal third-order methods were proposed for minimizing convex functions with Lipschitz third derivative. These developments are in large contrast to first-order methods, for which the worst-case function is quadratic [61] and has Lipschitz derivatives of any order, thus, preventing improvements under additional assumptions. This line of research was continued in [5], where such reductions were shown to be possible fo higher-order methods. We describe the ideas used in superfast methods in Section 4.

Tensor methods remain a very active area of research with many extensions of the existing methods. In particular, we mention adaptive variants for the setting when the Lipschitz constant is not known [41, 36], universal generalizations for the setting when the p-th derivative is Hölder continuous [33, 76, 19], versions with inexact minimization of the Taylor model [21, 35], tensor methods for finding approximate stationary points of convex functions [34, 23]. The ideas described above, especially the Monteiro–Svaiter accelerated proximal point method, turned out to be productive also in the areas not directly related to tensor methods, see non-trivial examples in [10, 12, 15]. Modern second-order and third-order methods demonstrate also their efficiency in Data Science and Machine Learning applications [18, 2, 24, 13]. We give more details of such applications in Section 5.

2 Notation and problem statement

We consider optimization problem

$$\min_{x \in \mathbb{R}^d} \{F(x) := f(x) + g(x)\}, \tag{1}$$

where f and g are convex functions. Denote x^* – the solution of the problem (1). If the solution is not unique we assume that x^* is such a solution that is the closest to starting point x_0 in 2-norm.

Denote $\|\cdot\|$ the Euclidean 2-norm in $\mathbb{R}^d$,

$$D^k f(x)[h]^k = \sum_{i_1,\ldots,i_d \geq 0:\ \sum_{j=1}^d i_j = k} \frac{\partial^k f(x)}{\partial x_1^{i_1} \ldots \partial x_d^{i_d}} h_1^{i_1} \cdot \ldots \cdot h_d^{i_d},$$

$$\|D^k f(x)\| = \max_{\|h\| \leq 1} \left\| D^k f(x)[h]^k \right\|.$$

Assume that f has Lipschitz derivatives of order p ($p \geq 1$):

$$\|D^p f(x) - D^p f(y)\| \leq L_{p,f} \|x - y\|, \ x, y \in \mathbb{R}^d. \tag{2}$$

Here and below (see e.g. (5)) we can consider that $x, y \in \mathbb{R}^d$ belongs to the ball in 2-norm centred at x^* with the radius $O(\|x_0 - x^*\|)$ [64].

The p-th Taylor polynomial of f is defined as

$$\Omega_p(f, x; y) = f(x) + \sum_{k=1}^{p} \frac{1}{k!} D^k f(x) \left[y - x\right]^k, \ y \in \mathbb{R}^d. \tag{3}$$

Note that from (2) it follows [63] that

$$\left| f(y) - \Omega_p(f, x; y) \right| \leq \frac{L_{p,f}}{(p+1)!} \|y - x\|^{p+1}. \tag{4}$$

We will say (see [70]) that F satisfies *r-growth condition* ($p + 1 \geq r \geq 1$) with constant $\sigma_r > 0$ iff

$$F(x) - F(x_*) \geq \sigma_r \|x - x^*\|^r, \ x \in \mathbb{R}^d. \tag{5}$$

Algorithm 1 Monteiro–Svaiter–Nesterov MSN(x_0,f,g,p,H,K)

Input: $p \geq 1$, $f : \mathbb{R}^d \to \mathbb{R}$, $g : \mathbb{R}^d \to \mathbb{R}$, $H > 0$.
$A_0 = 0, y_0 = x_0$.
for $k = 0$ **to** $k = K - 1$ **do**
  Find $\lambda_{k+1} > 0$ and $y_{k+1} \in \mathbb{R}^d$ such that

$$\frac{1}{2} \leq \lambda_{k+1} \frac{H\|y_{k+1} - \tilde{x}_k\|^{p-1}}{p!} \leq \frac{p}{p+1}, \text{ where}$$

$$y_{k+1} = \underset{y \in \mathbb{R}^d}{\operatorname{argmin}} \left\{ \Omega_p(f, \tilde{x}_k; y) + g(y) + \frac{H}{(p+1)!}\|y - \tilde{x}_k\|^{p+1} \right\}, \tag{6}$$

$$a_{k+1} = \frac{\lambda_{k+1} + \sqrt{\lambda_{k+1}^2 + 4\lambda_{k+1}A_k}}{2}, A_{k+1} = A_k + a_{k+1},$$

$$\tilde{x}_k = \frac{A_k}{A_{k+1}} y_k + \frac{a_{k+1}}{A_{k+1}} x_k.$$

  $x_{k+1} := x_k - a_{k+1}\nabla f(y_{k+1}) - a_{k+1}\nabla g(y_{k+1})$.
end for
return y_K

3 Optimal Tensor methods

The following algorithm is taken from [29], for $g \equiv 0$ see [11].

Theorem 1. *Let y_k be an output point of Algorithm 1 MSN(x_0, f, g, p, H, k) after k iterations, when $p \geq 1$ and $H \geq (p+1)L_{p,f}$. Then*

$$F(y_k) - F(x^*) \leq \frac{c_p H R^{p+1}}{k^{\frac{3p+1}{2}}}, \tag{7}$$

where $c_p = 2^{p-1}(p+1)^{\frac{3p+1}{2}}/p!$, $R = \|x_0 - x^*\|$.

Moreover, when $p \geq 2$ for ε: $F(y_k) - F(x_*) \leq \varepsilon$ it is required to solve auxiliary problem (6)*, to find (λ_{k+1}, y_{k+1}) with proper accuracy, $O\left(\ln\left(\varepsilon^{-1}\right)\right)$ times.*

Note that Theorem 1 is true also when $H \geq 2L_{p,f}$ (independently of $p \geq 1$). It can be derived from (4). The condition $H \geq (p+1)L_{p,f}$ was used only because it guarantees the convexity of auxiliary problem (6), see [63]. Under the conditions $H \geq (p+1)L_{p,f}$, $g \equiv 0$ for $p = 1, 2, 3$ there exist efficient ways to solve auxiliary problem (6), see [63]. For $p = 1$ there exists an explicit formula for the solution of (6). For $p = 2, 3$ the complexity of (6) is almost the same (up to a logarithmic factor in ε) as a complexity of Newton's method iteration [63], see also Section 4. It is important that there is no need to solve (6) accurately. It is sufficient to find such $\tilde{y}_{k+1}$ that satisfied

$$\left\|\nabla \widetilde{\Omega}^k(\tilde{y}_{k+1})\right\| \leq \frac{1}{4p(p+1)}\|\nabla F(\tilde{y}_{k+1})\|, \tag{8}$$

where

$$\widetilde{\Omega}^k(y) := \Omega_p(f, \tilde{x}_k; y) + g(y) + \frac{H}{(p+1)!}\|y - \tilde{x}_k\|^{p+1}.$$

Such a practical modification slow down the theoretical convergence rate in a factor $12/5$ in the right hand side of (7) [43, 35, 64].

In May 2022, D. Kovalev et al. [48] proposed for $g \equiv 0$ an explicit policy for choosing a pair (λ_{k+1}, y_{k+1}) in Algorithm 1 and modify the stopping criteria (8) according to [40]. It allows to solve (6) on each iteration at average only two times rather $O\left(\ln\left(\varepsilon^{-1}\right)\right)$ times as it was before. The final complexity bound for p-order oracle calls in [48] matched the lower oracle complexity bound from [63] (see also [26]) obtained for the worst-case function

$$F_p(x_1, ..., x_d) = |x_1|^{p+1} + |x_2 - x_1|^{p+1} + ... + |x_d - x_{d-1}|^{p+1}.$$

In concurrent and independent paper [14], the authors also proposed a way of reducing additional logarithmic factors. Note, that approach of [14] does not require any a priory knowledge of smoothness parameters (including Holder continuity assumption instead of Lipschitz one).

If additionally F satisfies r-growth condition (5) then optimal method can be developed based on restarts procedure [67, 28] – see Algorithm 2.

Algorithm 2 Restarted MSN(x_0,f,g,p,r,σ_r,H,K)

Input: $F = f + g : \mathbb{R}^d \to \mathbb{R}$ satisfies r-growth condition with constant σ_r, MSN(x_0,f,g,p,H,K).
$z_0 = x_0$.
for $k = 0$ **to** K **do**
  $R_k = R_0 \cdot 2^{-k}$,

$$N_k = \max\left\{\left\lceil\left(\frac{rc_pH2^r}{\sigma_r}R_k^{p+1-r}\right)^{\frac{2}{3p+1}}\right\rceil, 1\right\}. \tag{9}$$

  $z_{k+1} := y_{N_k}$, where y_{N_k} – output of MSN(z_k,f,g,p,H,N_k).
end for
return z_K

Theorem 2. *Let z_K be an output of Algorithm 2 after K restarts. If $H \geq (p+1)L_{p,f}$, $\sigma_r > 0$, then for $F(z_K) - F(x^*) \leq \varepsilon$ it sufficient to solve* (6)*:*

$$N = \tilde{O}\left(\left(\frac{HR^{p+1-r}}{\sigma_r}\right)^{\frac{2}{3p+1}}\right) \tag{10}$$

times, where $\tilde{O}(\,)$ – means the same as $O(\,)$ up to a $\ln^{\alpha}\left(\varepsilon^{-1}\right)$ factor.

Everything that was noted after Theorem 1 will also take place in this case.

In particular, when $g \equiv 0$, $p \geq 2$ and $r = 2$ we can replace MSN algorithm in Theorem 2 by Kovalev's variant of MSN [48] to obtain

$$O\left(\left(\frac{L_{p,f}R^{p-1}}{\sigma_2}\right)^{\frac{2}{3p+1}} + \log\log\left(\frac{\sigma_2^{\frac{p+1}{p-1}}}{L_{p,f}^{\frac{2}{p-1}}\varepsilon}\right)\right) \tag{11}$$

p-order oracle complexity bound. This upper bound corresponds to the σ_2-strongly convex case lower bound from [46]

and improves (10) on a logarithmic factor. The dependence on ε is $\log\log\varepsilon^{-1}$ as it should be locally for tensor methods [27, 20] (see also [58, 67, 4, 19]). But (11) describe two regimes: the first term (complexity independent on ε) describe the complexity to reach the vicinity of quadratic convergence, the second term (complexity is $\log\log\varepsilon^{-1}$) describe Newton's type local convergence.

Note that due to the presence of composite term g the described above MSN algorithm and its variations can be used for splitting oracle complexities [44, 47]. Namely, assume that f and g have Lipshitz p-th order derivatives with different Lipschitz constants. Based on MSN algorithm one can propose a general framework to accelerate tensor methods by splitting computational complexities. As a result, we can get near-optimal oracle complexity for each function in the sum separately for any $p \geq 1$, including the first-order methods. To be more precise, if the near optimal complexity to minimize f is $N_f(\varepsilon)$ and to minimize g is $N_g(\varepsilon)$, then MSN-based sliding algorithm [44, 47] requires no more than $\tilde{O}\left(N_f(\varepsilon)\right)$ oracle calls for f and $\tilde{O}\left(N_g(\varepsilon)\right)$ oracle calls for g to minimze $F = f + g$.

Not also that for $p = 1$ and $f \equiv 0$ we can take $H > 0$ in MSN and obtain Monteiro–Svaiter accelerated proximal envelop. In the cycle of recent papers [40, 16, 47, 49, 48] it was shown that this type of proximal (Catalyst-type [52]) envelop is logarithmic-free due to the well developed stopping rule criteria for the inner (auxiliary) problem. So it opens up different perspectives for improving existing Catalyst-based algorithms, see e.g. [78, 77].

4 Superfast acceleration and the structure of the auxiliary problem

In this section, we focus on the computationally efficient solution of the tensor subproblem for $p = 3$. With the proposed procedure, it is possible to implement third-order methods without the computation and storing of the third-order derivative. At the beginning, we show that this subproblem is convex. Then we move to the Bregman-distance gradient method as a subsolver for the tensor step. (6).

For this section, we are assuming that $g = 0$ and $p = 3$. Then $L_3 = L_{3,f}$, the third-order Taylor's polynomial is

$$\begin{aligned}\Omega(f, x; y) = \Omega_3(f, x; y) = f(x) + \nabla f(x)[y - x] \\ +\tfrac{1}{2}\nabla^2 f(x)\left[y - x\right]^2 + \tfrac{1}{6}D^3 f(x)\left[y - x\right]^3.\end{aligned} \tag{12}$$

The regularized third-order model is

$$\begin{aligned}\tilde{\Omega}(f, x; y) = f(x) + \nabla f(x)[y - x] \\ +\tfrac{1}{2}\nabla^2 f(x)\left[y - x\right]^2 + \tfrac{1}{6}D^3 f(x)\left[y - x\right]^3 + \tfrac{H}{24}\|y - x\|^4.\end{aligned} \tag{13}$$

The basic step for every tensor method is formulated as

$$x_{k+1} = \operatorname*{argmin}_{y\in\mathbb{R}^d}\left\{\tilde{\Omega}(f, x_k; y)\right\}. \tag{14}$$

This step is a major part of almost every third-order method. Next, we will describe how to effectively solve this subproblem without the computation of the third-order derivative and only using second-order information.

First, we clarify that the function (13) is convex for $H \geq 3L_3$. Note, that if $H \geq L_3$ the model (13) is possibly non-convex because of third-order derivative. It means that the minimizing non-convex subproblem is harder than minimizing original convex problem. In 2018 Yu. Nesterov made the breakthrough in [63]. He proved that if $H \geq 3L_3$ then the model (13) is convex and high-order methods are implementable.

To give some intuition, we present a sketch of the convexity proof. The following inequality can be derived from the Lipschitz-continuous condition and the convexity of function f.

$$0 \leq \nabla^2 f(y) \leq \nabla^2\Omega(f, x; y) + \frac{L_3}{2}\|y - x\|^2 I.$$

Now, to finish the proof, we need to choose H such that

$$\nabla^2\tilde{\Omega}(f, x; y) \geq \nabla^2\Omega(f, x; y) + \frac{L_3}{2}\|y - x\|^2 I.$$

This lead us to the crucial detail, that was misleading before

$$\nabla^2\left\{\tfrac{1}{4}\|x\|^4\right\} = 2xx^T + \|x\|^2 I \geq \|x\|^2 I.$$

So, from the fact that x is a vector and xx^T is a singular matrix, we are losing factor 3 in the last inequality. Finally, we get that if $H \geq 3L_3$ then the function (13) is convex. For more details one can check Theorem 1 in [63].

Thus, we can move to the effective subsolver of the problem (14) by Bregman-distance gradient method for relatively smooth functions from [54]. The main idea is to show that the optimized function $\phi(y)$ is L_ρ-smooth and μ_ρ-strongly convex with respect to some convex function $\rho(y)$ or

$$\mu_\rho\nabla^2\rho(y) \leq \nabla^2\phi(y) \leq L_\rho\nabla^2\rho(y).$$

Then Bregman-distance gradient method make next steps

$$y_{t+1} = \operatorname*{argmin}_{y\in\mathbb{R}^d}\left\{\langle\nabla\phi(y_t), y - y_t\rangle + L_\rho\beta_\rho\left(y_t, y\right)\right\},$$

where

$$\beta_\rho\left(y_t, y\right) = \rho(y) - \rho(y_t) - \langle\nabla\rho(y_t), y - y_t\rangle$$

is a Bregman-distance generated by $\rho(y)$. Bregman-distance gradient method converges linearly with condition number $\kappa = \frac{L_\rho}{\mu_\rho}$ and convergence rate

$$N = \tfrac{L_\rho}{\mu_\rho}\log\left(\tfrac{L_\rho\beta_\rho(y_0, y_*)}{\varepsilon}\right).$$

One can show that the model (13) with $H = 6L_3$ can be optimized as $\phi(y) = \tilde{\Omega}(f, x_k; y)$ by gradient method with Bregman-distance generated by

$$\rho_{x_k}(y) = \tfrac{1}{2}\nabla^2 f(x_k)\left[y - x_k\right]^2 + \tfrac{L_3}{4}\|y - x_k\|^4,$$

$L_\rho = 1 + \frac{1}{\sqrt{2}}$, $\mu_\rho = 1 - \frac{1}{\sqrt{2}}$, and $\kappa = \frac{1}{\left(1+\sqrt{2}\right)^2} = O(1)$. This means that this method is very fast and converges with a fixed number of iterations. It is worth noting that for each step, the full hessian for β is computed, but the full third-order derivative is unnecessary because we only need the derivative-vector-vector product $D^3 f(x_k)\left[y - x\right]^2$ to compute $\nabla\tilde{\Omega}(f, x_k; y)$. Derivative-vector-vector product can be efficiently and precisely computed by autogradient or approximated by finite difference.

To summarize, we get the tensor method with a convergence rate of $1/k^5$ for third-order methods but only the first and second-order derivatives are computed. This approach was firstly proposed in [63] and then it was improved in [65]. Section 5 from [63] is good for a general understanding of this approach. For details and precise proofs, one can read [65].

5 Tensor methods and stochastic distributed setup

As well as first-order methods, modifications of tensor methods are proposed for problems of the finite sum type, stochastic and distributed optimization.

5.1 Stochastic Optimization

In this subsection, we consider a problem (1) with $g = 0$ under inexact information on the derivatives of objective f up to order $p \geq 2$. In particular, we are motivated by stochastic optimization. In this case, objective f has the following form

$$f(x) = \mathbb{E}_{\xi \sim \mathcal{D}}[f(x;\xi)], \tag{15}$$

where the random variable ξ is sampled from a distribution $\mathcal{D}$, and an optimization procedure has access only to stochastic realizations of the function $f(x;\xi)$ via its higher-order derivatives up to the order $p \geq 2$.

The full Taylor expansion of f (3) requires computing all the derivatives up to the order p, which can be expensive to calculate. Following the works [32, 3] let us use some approximations $G_{x,i}$ for the derivatives $\nabla^i f(x)$, $i = 1, \ldots, p$ and construct an inexact p-th order Taylor expansion of the objective:

$$\Phi_p(f, x; y) = f(x) + \sum_{i=1}^{p} \frac{1}{i!} G_{x,i}[y - x]^i, \tag{16}$$

where $G_{x,i}$ satisfies the following

Condition 1. *Given the inexactness levels $\delta_i \geq 0$ for $i = 1, \ldots, p$, for all $x \in \mathbb{R}^n$ the approximate derivatives $G_{x,i}$ satisfy for all $y \in \mathbb{R}^n$ the following inequalities:*

$$\|(G_{x,i} - \nabla^i f(x))[y - x]^{i-1}\| \leq \delta_i \|y - x\|^{i-1}, i = 1, \ldots, p. \tag{17}$$

Next, we describe inexact tensor methods under Condition 1. Then, motivated by stochastic optimization (15), we focus on stochastic approximation of derivatives via sampling.

Recall, that exact tensor methods are based on minimisation of tensor model $\tilde{\Omega}(f, x; y)$ (see eq. (13) for $p = 3$). For inexact tensor methods the model is constructed in the following way [3]:

$$\tilde{\Phi}_p(f, x; y) = \Phi_p(f, x; y) + \\ + \sum_{i=1}^{p} \frac{\delta_i}{i(i-2)!} \|y - x\|^i + \frac{H}{(p+1)(p-1)!} \|y - x\|^{p+1}.$$

This model satisfies two main conditions ([3], Theorem 1):

- Model $\tilde{\Phi}_p(f, x; y)$ is a global upper bound for the function f:
$$f(y) \leq \tilde{\Phi}_p(f, x; y), \ x, y \in \mathbb{R}^n. \tag{18}$$
- Model $\tilde{\Phi}_p(f, x; y)$ is convex.

Based on these properties we can define the step of the Inexact Tensor Method as

$$x_{t+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \tilde{\Phi}_p(f, x_t; y). \tag{19}$$

Note, that in the view of inequality (18) this method is monotone. Accelerated inexact tensor methods were also proposed in [32, 3]. They are based on the same idea but utilise some acceleration schemes.

The Inexact Tensor Method (eq. (19)) has the following convergence [3]:

$$f(x_T) - f(x^*) \leq \\ O\left(\sum_{i=1}^{p} \frac{\delta_i}{T^{i-1}} \max_{x \in \mathcal{L}(x_0)} \|x - x^*\|^i + \frac{L_p}{T^p} \max_{x \in \mathcal{L}(x_0)} \|x - x^*\|^{p+1} \right),$$

where $\mathcal{L}(x_0) = \{x | f(x) \leq f(x_0)\}$, L_p is the Lipschitz constant of the p-th derivative, and T is the iteration counter. For the Accelerated Inexact Tensor Method the following convergence rate holds:

$$f(x_T) - f(x^*) \leq O\left(\delta_1 \bar{R} + \sum_{i=2}^{p} \frac{\delta_i}{T^i} \|x^* - x_0\|^i + \frac{L_p}{T^{p+1}} \|x^* - x_0\|^{p+1} \right),$$

where $\bar{R}$ is such a constant that $\|x_t - x^*\| \leq \bar{R}$ and $\|x_1 - x_0\| \leq \bar{R}$.

Stochastic Tensor Method and Accelerated Stochastic Tensor Method were proposed as a particular case of general inexact tensor methods in [32] for $p = 2$ and in [3] for $p \geq 2$ (see [8, 55] for non-convex problems). In each iteration of stochastic tensor methods, we sample stochastic derivatives to form mini-batch approximations for the derivatives of f. More precisely, for $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_p$ being sample sets for each derivative, we set

$$G_{x,i} = \frac{1}{|\mathcal{S}_i|} \sum_{j \in \mathcal{S}_i} \nabla^i f\left(x, \xi_j\right), \quad i = 1, .., p. \tag{20}$$

Next, we provide theoretical guarantees on sample sizes $|\mathcal{S}_i|$, so the convergence rate of stochastic tensor methods matches exact versions up to a constant. We underline that the results for stochastic optimization problems are corollaries of the analysis [3] of Inexact Tensor Method and Inexact Accelerated Tensor Method.

Theorem 3. *Assume that the he function $f(x)$ and its derivatives $\nabla f(x), \ldots, \nabla^p f(x)$ are Lipschitz continuous:*

$$\|\nabla^i f(x) - \nabla^i f(y)\| \leq L_i \|x - y\|, i = 0, \ldots, p$$

and for all $i = 1, 2, \ldots, p$, ξ, and $x \in \mathbb{R}^n$:

$$\|\nabla^i f(x, \xi) - \nabla^i f(x)\| \leq M_i. \tag{21}$$

Let $D = \max_{x \in \mathcal{L}(x_0)} \|x - x^*\|$, $R = \|x^* - x_0\|$ and ε is a desired accuracy after T iterations: $f(x^T) - f(x^*) \leq \varepsilon$.

- *Then, we can choose the sizes $|\mathcal{S}_i|$ of sample sets $\mathcal{S}_i$ in (20) to be*
$$n_i := |\mathcal{S}_1| = O\left(\frac{(M_1 + L_0)^2}{(L_p + pH)^{\frac{2(i-1)}{p}}} \left(\frac{\varepsilon}{D}\right)^{-2} \log\left(\frac{1}{\delta}\right) \right), \ i = 1 \ldots p$$
*so that with probability at least $1 - \delta$ **Stochastic Tensor Method** has complexity $T = O\left(\frac{L_p^{1/p} D^{(p+1)/p}}{\varepsilon^{1/p}} \right)$.*

- *Then, we can choose the sizes $|\mathcal{S}_i|$ of sample sets $\mathcal{S}_i$ in* (20) *to be*

$$n_1 = |\mathcal{S}_1| = O\left((L_0 + M_1)^2 \left(\frac{\varepsilon}{R}\right)^{-2} \log\left(\frac{1}{\delta}\right)\right),$$

$$n_i = |\mathcal{S}_i| = O\left(\frac{(L_{i-1} + M_i)^2}{(L_p + pH)^{\frac{2i}{p+1}}} \cdot \varepsilon^{-2\frac{p-i+1}{p+1}} \log\left(\frac{1}{\delta}\right)\right), i = 2, \ldots, p$$

so that with probability at least $1-\delta$ ***Accelerated Stochastic Tensor Method*** *has complexity* $T = O\left(\frac{L_p^{1/(p+1)}R}{\varepsilon^{1/(p+1)}}\right)$.

Theorem 3 shows how to choose batch sizes for derivatives in inexact tensor methods to achieve convergence as in exact case for online setting. Gradient sample sizes of non-accelerated and accelerated tensor methods are the same. For the case of higher derivatives (of order ≥ 2), we need to use fewer samples for the accelerated method than for the non-accelerated one. Moreover, batch sizes are decreasing with the growth of derivatives order.

Note that the finite sum problem $\min\limits_{x\in\mathbb{R}^d}\left\{f(x) = \frac{1}{m}\sum\limits_{i=1}^{m} f_i(x)\right\}$ can be considered as a particular case of the problem (15) if we set ξ to be uniformly distributed over $i = 1, ..., m$. But the theoretical analysis in this can be made under weaker assumptions: eq. (21) is not needed.

5.2 Distributed optimization

In this subsection, we consider a distributed empirical risk minimization problem. In this setup, multiple agents/devices/workers collaborate to solve a machine learning problem by communicating over a central node (server). The goal is to minimize the following finite-sum objective:

$$\min_{x\in\mathbb{R}^d}\left\{f(x) := \frac{1}{m}\sum_{j=1}^{m} f_j(x)\right\}, \tag{22}$$

where $f_j(x)$ is the loss function (parametrized by $x \in \mathbb{R}^d$) associated with the data stored on the j-th machine and the μ_f-strongly convex function $f(x)$ is the global empirical loss function. Each node has access only to its local data and can communicate with the central server.

Problems of the form (22) arise in distributed machine learning [50, 51] and other applications [80, 72, 57, 22, 79, 73]. Beyond first-order methods, second-order methods have been developed for such Federated Learning problems. One of the keys to efficient second-order methods for this setting is exploiting the statistical similarity of the data [81, 24, 18, 13, 2]. Using statistical arguments, it is possible to show that if $f_1(x) = \frac{1}{n}\sum_{l=1}^{n} \ell(x, y_l)$, where $\ell(x, y_l)$ is the individual loss function of each example l stored at the first device chosen to be the central device, then $\|\nabla^2 f(x) - \nabla^2 f_1(x)\|_2 \leq \sigma$, where σ is proportional to $\frac{1}{\sqrt{n}}$. Then, defining

$$\rho(x) = \frac{1}{n}\sum_{l=1}^{n} \ell(x; y_l) + \frac{\sigma}{2}\|x\|_2^2, \tag{23}$$

we obtain that the global objective f is strongly convex and smooth relative to ρ, i.e., (cf. Section 4)

$$\mu_\rho \nabla^2\phi(x) \leq \nabla^2 f(x) \leq L_\rho \nabla^2\phi(x),$$

where $L_\rho = 1$, $\mu_\rho = \mu_f/(\mu_f + 2\sigma)$, and $\kappa_\rho = L_\rho/\mu_\rho = 1 + 2\sigma/\mu_f$. Thus, as in Section 4 one can apply the Bregman-distance gradient method and its «accelerated» variant [38].

When implemented, these methods require to minimize at master node (the first node)

$$\min_y \left\{\langle \nabla f(y_m), y - y_m\rangle + L_\phi \beta_\rho(y_m, y)\right\}.$$

The first term is available due to communications and the last term is a sum-type function with n terms according to (23). If n is large enough and d is moderate then Hessian calculation time can dominate Hessian inversion time. That allows to use efficiently Tensor methods. In particular Superfast second-order methods mentioned in the Section 4.

This idea was exploited and analysed in [24] which resulted in an efficient distributed second-order solver with good practical performance and in a certain regime total arithmetic operations complexity being better than that of the existing variance reduced first-order algorithms for problem (22).

The work of A. Agafonov was supported by the Ministry of Science and Higher Education of the Russian Federation (Goszadaniye), No. 075-00337-20-03, project No. 0714-2020-0005. The work of A. Gasnikov was supported by the strategic academic leadership program «Priority 2030» (Agreement 075-02-2021-1316 30.09.2021).

References

[1] Deeksha Adil, Brian Bullins, Arun Jambulapati, and Sushant Sachdeva. Line search-free methods for higher-order smooth monotone variational inequalities. *arXiv preprint arXiv:2205.06167*, 2022.

[2] Artem Agafonov, Pavel Dvurechensky, Gesualdo Scutari, Alexander Gasnikov, Dmitry Kamzolov, Aleksandr Lukashevich, and Amir Daneshmand. An accelerated second-order method for distributed stochastic optimization. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 2407–2413. IEEE, 2021.

[3] Artem Agafonov, Dmitry Kamzolov, Pavel Dvurechensky, and Alexander Gasnikov. Inexact tensor methods and their application to stochastic convex optimization. *arXiv preprint arXiv:2012.15636*, 2020.

[4] Naman Agarwal and Elad Hazan. Lower bounds for higher-order convex optimization. In *Conference On Learning Theory*, pages 774–792. PMLR, 2018.

[5] Masoud Ahookhosh and Yurii Nesterov. High-order methods beyond the classical complexity bounds, ii: inexact high-order proximal-point methods with segment search. *arXiv preprint arXiv:2109.12303*, 2021.

[6] Yossi Arjevani, Ohad Shamir, and Ron Shiff. Oracle complexity of second-order methods for smooth convex optimization. *Mathematical Programming*, 178(1):327–360, 2019.

[7] Michel Baes. Estimate sequence methods: extensions and approximations. *Institute for Operations Research, ETH, Zürich, Switzerland*, 2009.

[8] Stefania Bellavia, Gianmarco Gurioli, Benedetta Morini, and Philippe L. Toint. Adaptive regularization algorithms with inexact evaluations for nonconvex optimization. *SIAM Journal on Optimization*, 29(4):2881–2915, 2019.

[9] Ali Bouaricha. Tensor methods for large, sparse unconstrained optimization. *SIAM Journal on Optimization*, 7(3):732–756, 1997.

[10] Sébastien Bubeck, Qijia Jiang, Yin-Tat Lee, Yuanzhi Li, and Aaron Sidford. Complexity of highly parallel non-smooth convex optimization. *Advances in Neural Information Processing Systems*, 32, 2019.

[11] Sébastien Bubeck, Qijia Jiang, Yin Tat Lee, Yuanzhi Li, and Aaron Sidford. Near-optimal method for highly smooth convex optimization. In *Conference on Learning Theory*, pages 492–507. PMLR, 2019.

[12] Brian Bullins. Highly smooth minimization of non-smooth problems. In *Conference on Learning Theory*, pages 988–1030. PMLR, 2020.

[13] Brian Bullins, Kshitij Patel, Ohad Shamir, Nathan Srebro, and Blake E Woodworth. A stochastic newton algorithm for distributed convex optimization. *Advances in Neural Information Processing Systems*, 34:26818–26830, 2021.

[14] Yair Carmon, Danielle Hausler, Arun Jambulapati, Yujia Jin, and Aaron Sidford. Optimal and adaptive monteiro-svaiter acceleration. *arXiv preprint arXiv:2205.15371*, 2022.

[15] Yair Carmon, Arun Jambulapati, Yujia Jin, and Aaron Sidford. Thinking inside the ball: Near-optimal minimization of the maximal loss. In *Conference on Learning Theory*, pages 866–882. PMLR, 2021.

[16] Yair Carmon, Arun Jambulapati, Yujia Jin, and Aaron Sidford. Recapp: Crafting a more efficient catalyst for convex optimization. In *International Conference on Machine Learning*, pages 2658–2685. PMLR, 2022.

[17] Pafnuty Chebyshev. *Collected works. Vol 5*. Strelbytskyy Multimedia Publishing, 2018.

[18] Amir Daneshmand, Gesualdo Scutari, Pavel Dvurechensky, and Alexander Gasnikov. Newton method over networks is fast up to the statistical precision. In *International Conference on Machine Learning*, pages 2398–2409. PMLR, 2021.

[19] Nikita Doikov, Konstantin Mishchenko, and Yurii Nesterov. Super-universal regularized newton method. *arXiv preprint arXiv:2208.05888*, 2022.

[20] Nikita Doikov and Yurii Nesterov. Local convergence of tensor methods. *Mathematical Programming*, 193(1):315–336, 2022.

[21] Nikita Doikov and Yurii E Nesterov. Inexact tensor methods with dynamic accuracies. In *ICML*, pages 2577–2586, 2020.

[22] Pavel Dvurechenskii, Darina Dvinskikh, Alexander Gasnikov, Cesar Uribe, and Angelia Nedich. Decentralize and randomize: Faster algorithm for wasserstein barycenters. In *Advances in Neural Information Processing Systems*, pages 10760–10770, 2018.

[23] Pavel Dvurechensky, Alexander Gasnikov, Petr Ostroukhov, César A. Uribe, and Anastasiya Ivanova. Near-optimal tensor methods for minimizing the gradient norm of convex function. *arXiv:1912.03381*, 2019. WIAS Preprint No. 2694.

[24] Pavel Dvurechensky, Dmitry Kamzolov, Aleksandr Lukashevich, Soomin Lee, Erik Ordentlich, César A Uribe, and Alexander Gasnikov. Hyperfast second-order local solvers for efficient statistically preconditioned distributed optimization. *arXiv preprint arXiv:2102.08246*, 2021.

[25] Yurii Gavrilovich Evtushenko and Alexey Anatol'evich Tret'yakov. pth-order approximation of the solution set of nonlinear equations. *Computational Mathematics and Mathematical Physics*, 53(12):1763–1780, 2013.

[26] Ankit Garg, Robin Kothari, Praneeth Netrapalli, and Suhail Sherif. Near-optimal lower bounds for convex optimization for all orders of smoothness. *Advances in Neural Information Processing Systems*, 34:29874–29884, 2021.

[27] Alexander Gasnikov. Modern numerical methods. universal gradient descent. *MIPT, arXiv:1711.00394*, 2018.

[28] Alexander Gasnikov, Pavel Dvurechensky, Eduard Gorbunov, Evgeniya Vorontsova, Daniil Selikhanovych, and César A Uribe. Optimal tensor methods in smooth convex and uniformly convexoptimization. In *Conference on Learning Theory*, pages 1374–1391. PMLR, 2019.

[29] Alexander Vladimirovich Gasnikov, Darina Mikhailovna Dvinskikh, Pavel Evgenievich Dvurechensky, Dmitry Igorevich Kamzolov, Vladislav Vyacheslavovich Matyukhin, Dmitry Arkadievich Pasechnyuk, Nazarii Konstantinovich Tupitsa, and Aleksey Vladimirovich Chernov. Accelerated meta-algorithm for convex optimization problems. *Computational Mathematics and Mathematical Physics*, 61(1):17–28, 2021.

[30] Alexander Vladimirovich Gasnikov and Dmitry Alexandrovich Kovalev. A hypothesis about the rate of global convergence for optimal methods (newton's type) in smooth convex optimization. *Computer research and modeling*, 10(3):305–314, 2018.

[31] A.V. Gasnikov, E.A. Gorbunov, D.A. Kovalev, A.A.M. Mohammed, and E.O. Chernousova. Substantiation of the hypothesis about optimal estimates of the rate of convergence of numerical methods of high-order convex optimization. *Computer Research and Modeling*, 10(6):737–753, 2018.

[32] Saeed Ghadimi, Han Liu, and Tong Zhang. Second-order methods with cubic regularization under inexact information. *arXiv preprint arXiv:1710.05782*, 2017.

[33] G. N. Grapiglia and Yu. Nesterov. Tensor methods for minimizing convex functions with hölder continuous higher-order derivatives. *SIAM Journal on Optimization*, 30(4):2750–2779, 2020.

[34] G. N. Grapiglia and Yurii Nesterov. Tensor methods for finding approximate stationary points of convex functions. *Optimization Methods and Software*, 0(0):1–34, 2020.

[35] Geovani Nunes Grapiglia and Yurii Nesterov. On inexact solution of auxiliary problems in tensor methods for convex optimization. *Optimization Methods and Software*, 36(1):145–170, 2021.

[36] Geovani Nunes Grapiglia and Yurii Nesterov. Adaptive third-order methods for composite convex optimization. *arXiv:2202.12730*, 2022.

[37] Andreas Griewank. The modification of newton's method for unconstrained optimization by bounding cubic terms. Technical report, Technical report NA/12, 1981.

[38] Hadrien Hendrikx, Lin Xiao, Sebastien Bubeck, Francis Bach, and Laurent Massoulie. Statistically preconditioned accelerated gradient method for distributed optimization. In *International conference on machine learning*, pages 4203–4227. PMLR, 2020.

[39] Karl-Heinz Hoffmann and Hans-Joachim Kornstaedt. Higher-order necessary conditions in abstract mathematical programming. *Journal of Optimization Theory and Applications*, 26(4):533–568, 1978.

[40] Anastasiya Ivanova, Dmitry Pasechnyuk, Dmitry Grishchenko, Egor Shulgin, Alexander Gasnikov, and Vladislav Matyukhin. Adaptive catalyst for smooth convex optimization. In *International Conference on Optimization and Applications*, pages 20–37. Springer, 2021.

[41] Bo Jiang, Tianyi Lin, and Shuzhong Zhang. A unified adaptive tensor approximation scheme to accelerate composite convex optimization. *SIAM Journal on Optimization*, 30(4):2897–2926, 2020.

[42] Bo Jiang, Haoyue Wang, and Shuzhong Zhang. An optimal high-order tensor method for convex optimization. In *Conference on Learning Theory*, pages 1799–1801. PMLR, 2019.

[43] Dmitry Kamzolov and Alexander Gasnikov. Near-optimal hyperfast second-order method for convex optimization and its sliding. *arXiv preprint arXiv:2002.09050*, 2020.

[44] Dmitry Kamzolov, Alexander Gasnikov, and Pavel Dvurechensky. Optimal combination of tensor optimization methods. In *International Conference on Optimization and Applications*, pages 166–183. Springer, 2020.

[45] Leonid Vital'evich Kantorovich. On newton's method. *Trudy Matematicheskogo Instituta imeni VA Steklova*, 28:104–144, 1949.

[46] Guy Kornowski and Ohad Shamir. High-order oracle complexity of smooth and strongly convex optimization. *arXiv preprint arXiv:2010.06642*, 2020.

[47] Dmitry Kovalev, Aleksandr Beznosikov, Ekaterina Borodich, Alexander Gasnikov, and Gesualdo Scutari. Optimal gradient sliding and its application to distributed optimization under similarity. *arXiv preprint arXiv:2205.15136*, 2022.

[48] Dmitry Kovalev and Alexander Gasnikov. The first optimal acceleration of high-order methods in smooth convex optimization. *arXiv preprint arXiv:2205.09647*, 2022.

[49] Dmitry Kovalev and Alexander Gasnikov. The first optimal acceleration of high-order methods in smooth convex optimization. *arXiv preprint arXiv:2205.09647*, 2022.

[50] Tim Kraska, Ameet Talwalkar, John C Duchi, Rean Griffith, Michael J. Franklin, and Michael I. Jordan. Mlbase: A distributed machine-learning system. In *CIDR*, volume 1, pages 2–1, 2013.

[51] Mu Li, David G Andersen, Jun Woo Park, Alexander J Smola, Amr Ahmed, Vanja Josifovski, James Long, Eugene J Shekita, and Bor-Yiing Su. Scaling distributed machine learning with the parameter server. In *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14)*, pages 583–598, 2014.

[52] Hongzhou Lin, Julien Mairal, and Zaid Harchaoui. Catalyst acceleration for first-order convex optimization: from theory to practice. *Journal of Machine Learning Research*, 18(1):7854–7907, 2018.

[53] Tianyi Lin, Michael Jordan, et al. Perseus: A simple high-order regularization method for variational inequalities. *arXiv preprint arXiv:2205.03202*, 2022.

[54] Haihao Lu, Robert M Freund, and Yurii Nesterov. Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018.

[55] Aurelien Lucchi and Jonas Kohler. A stochastic tensor method for non-convex optimization. *arXiv preprint arXiv:1911.10367*, 2019.

[56] Renato D.C. Monteiro and Benar Fux 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.

[57] Angelia Nedić, Alex Olshevsky, and César A Uribe. Fast convergence rates for distributed non-bayesian learning. *IEEE Transactions on Automatic Control*, 62(11):5538–5553, 2017.

[58] Arkadij Semenovič Nemirovskij and David Borisovich Yudin. Problem complexity and method efficiency in optimization. 1983.

[59] Yu Nesterov. Accelerating the cubic regularization of newton's method on convex problems. *Mathematical Programming*, 112(1):159–181, 2008.

[60] 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.

[61] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

[62] Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

[63] Yurii Nesterov. Implementable tensor methods in unconstrained convex optimization. *Mathematical Programming*, 186(1):157–183, 2021.

[64] Yurii Nesterov. Inexact high-order proximal-point methods with auxiliary search procedure. *SIAM Journal on Optimization*, 31(4):2807–2828, 2021.

[65] Yurii Nesterov. Superfast second-order methods for unconstrained convex optimization. *Journal of Optimization Theory and Applications*, 191(1):1–30, 2021.

[66] Yurii Nesterov and Arkadii Nemirovskii. *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.

[67] Yurii Nesterov and Boris Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

[68] Isaac Newton. Methodus fluxionum et serierum infinitarum, 1671. *The method of fluxions and infinite series*, 1967.

[69] Jorge Nocedal and Stephen J. Wright. *Numerical optimization*. Springer, 1999.

[70] Boris Teodorovich Poljak. Sharp minimum. In *Generalized Lagrangians and applications*. Oxford: Pergamon Press, 1982.

[71] B. Polyak. Introduction to optimization. *Optimization Software*, 1987.

[72] M.G. Rabbat and R.D. Nowak. Decentralized source localization and tracking wireless sensor networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 921–924, 2004.

[73] Sundhar Srinivasan Ram, Venugopal V Veeravalli, and Angelia Nedic. Distributed non-autonomous power control through distributed convex optimization. In *IEEE INFOCOM 2009*, pages 3001–3005. IEEE, 2009.

[74] Robert B Schnabel and Ta-Tung Chow. Tensor methods for unconstrained optimization using second derivatives. *SIAM Journal on Optimization*, 1(3):293–315, 1991.

[75] Robert B. Schnabel and Paul D. Frank. Tensor methods for nonlinear equations. *SIAM Journal on Numerical Analysis*, 21(5):815–843, 1984.

[76] Chaobing Song, Yong Jiang, and Yi Ma. Unified acceleration of high-order algorithms under general hölder continuity. *SIAM Journal on Optimization*, 31(3):1797–1826, 2021.

[77] Ye Tian, Gesualdo Scutari, Tianyu Cao, and Alexander Gasnikov. Acceleration in distributed optimization under similarity. In *International Conference on Artificial Intelligence and Statistics*, pages 5721–5756. PMLR, 2022.

[78] Vladislav Tominin, Yaroslav Tominin, Ekaterina Borodich, Dmitry Kovalev, Alexander Gasnikov, and Pavel Dvurechensky. On accelerated methods for saddle-point problems with composite structure. *arXiv preprint arXiv:2103.09344*, 2021.

[79] César A. Uribe, Darina Dvinskikh, Pavel Dvurechensky, Alexander Gasnikov, and Angelia Nedić. Distributed computation of Wasserstein barycenters over networks. In *2018 IEEE 57th Annual Conference on Decision and Control (CDC)*, 2018. Accepted, arXiv:1803.02933.

[80] Lin Xiao and Stephen Boyd. Optimal scaling of a gradient method for distributed resource allocation. *Journal of Optimization Theory and Applications*, 129(3):469–488, 2006.

[81] Yuchen Zhang and Lin Xiao. *Communication-Efficient Distributed Optimization of Self-concordant Empirical Loss*, pages 289–341. Springer International Publishing, Cham, 2018.