# Spectral Preconditioning for Gradient Methods on Graded Non-convex Functions

**Nikita Doikov** [1] **Sebastian U. Stich** [2] **Martin Jaggi** [1]

## Abstract

The performance of optimization methods is often tied to the spectrum of the objective Hessian. Yet, conventional assumptions, such as smoothness, do often not enable us to make finely-grained convergence statements—particularly not for non-convex problems. Striving for a more intricate characterization of complexity, we introduce a unique concept termed *graded non-convexity*. This allows to partition the class of non-convex problems into a nested chain of subclasses. Interestingly, many traditional non-convex objectives, including partially convex problems, matrix factorizations, and neural networks, fall within these subclasses. As a second contribution, we propose gradient methods with spectral preconditioning, which employ inexact top eigenvectors of the Hessian to address the ill-conditioning of the problem, contingent on the grade. Our analysis reveals that these new methods provide provably superior convergence rates compared to basic gradient descent on applicable problem classes, particularly when large gaps exist between the top eigenvalues of the Hessian. Our theory is validated by numerical experiments executed on multiple practical machine learning problems.

## 1. Introduction

**Motivation.** The gradient method is an important and attractive tool for solving large-scale optimization problems. It has very cheap cost of every iteration and well established convergence guarantees, that hold starting from an arbitrary initial point and for a wide family of problem classes, including convex and non-convex problems. However, the major drawback of the gradient method remains to be its slow rate of convergence: for solving modern optimization problems up to a reasonable accuracy level, it is often required to do a lot of gradient steps, due to *ill-conditioning* of the problem.

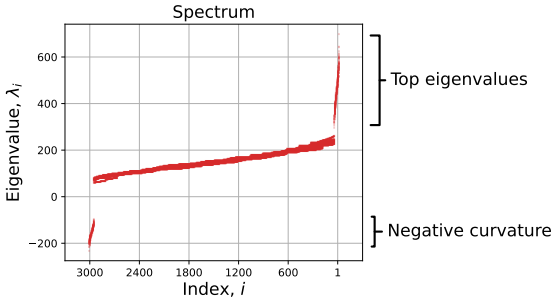


*Figure 1.* Spectrum of the Hessians for the matrix factorization problem (Example 2.10) of the optimization dimension $n = 3000$, for 10 random objectives.

In order to improve the gradient direction, we can multiply the gradient by a specifically crafted matrix called *preconditioner*, which should adjust the method to the right geometry of the problem. However, finding a good preconditioner with strong theoretical guarantees is not easy, especially for non-convex problems. In this work, we propose a new family of *spectral preconditioners* that rely on an additional refined information about the function class. As a by-product, we establish convergence rates, that are provably better than those of the gradient methods.

Optimization theory suggests that the main complexity parameters that affect the rate of convergence for gradient methods are the spectrum of the Hessian $\nabla^2 f$ and its extremal characteristics, such as the bound for the *maximal eigenvalue* (the Lipschitz constant of the gradient) or the *condition number* (the ratio of the largest and smallest eigenvalues) (Nemirovski & Yudin, 1983; Nesterov, 2018). Moreover, some of the fundamental properties of the objective function, such as *convexity*, *weak convexity*, or *strong convexity*, that distinguish between all optimization problems and globally solvable ones, can be defined in terms of lower bounds on the spectrum. Thus (for a twice differentiable function),

$$f \text{ is convex} \quad \Leftrightarrow \quad \nabla^2 f \succeq \mathbf{0}. \tag{1}$$

[1]Machine Learning and Optimization Laboratory (MLO), EPFL, Lausanne, Switzerland [2]CISPA Helmholtz Center for Information Security, Saarbrücken, Germany. Correspondence to: Nikita Doikov <nikita.doikov@epfl.ch>, Sebastian U. Stich <stich@cispa.de>, Martin Jaggi <martin.jaggi@epfl.ch>.

*Proceedings of the $41^{st}$ International Conference on Machine Learning*, Vienna, Austria. PMLR 235, 2024. 

| Algorithm | Preconditioning | Non-convex Complexity | Strongly Convex | Arithmetic Cost |
|---|---|---|---|---|
| The Gradient Method | $\mathbf{H} := \mathbf{I}, \quad \tau = 0$ | $\mathcal{O}\left(\frac{\lambda_1}{\varepsilon^2}\right) \cdot F_0$ | $\tilde{\mathcal{O}}\left(\frac{\lambda_1}{\lambda_n}\right)$ | $\mathcal{O}(n)$ |
| Spectral Preconditioning **(ours)** | $\mathbf{H} \approx \nabla^2_\tau f, \quad 1 \leq \tau < n$ | $\mathcal{O}\left(\frac{\lambda_{\tau+1}}{\varepsilon^2} + \frac{L^{1/2}}{\varepsilon^{3/2}}\right) \cdot F_0$ | $\tilde{\mathcal{O}}\left(\frac{\lambda_{\tau+1}}{\lambda_n} + MD\right)$ | $\mathcal{O}(\tau^2 n + \tau^3)$ |
| Newton's Method | $\mathbf{H} := \nabla^2 f, \quad \tau = n$ | $\mathcal{O}\left(\frac{L^{1/2}}{\varepsilon^{3/2}}\right) \cdot F_0$ | $\tilde{\mathcal{O}}(MD)$ | $\mathcal{O}(n^3)$ |

*Table 1.* Global complexity bounds for our method as compared to the classic gradient method and the regularized Newton methods. We denote by $\lambda_i$ a uniform bound for the $i$th eigenvalue of the Hessian, sorted in a non-ascending order: $\lambda_1 \geq \ldots \geq \lambda_n$. We denote by $L$ the Lipschitz constant of the Hessian and, for convex objectives, $M$ is the constant of quasi-self-concordance (see Section 6). $F_0 := f(\boldsymbol{x}_0) - f^\star$, and $D$ is the diameter of the initial sublevel set. We see that using the spectral preconditioner of order $\tau$, we cut the top $\tau$ eigenvalues of the spectrum, which is the most significant complexity factor. We present the state-of-the-art global complexities from (Nesterov & Polyak, 2006; Doikov, 2023) for Newton's Method with cubic and gradient regularizations.

At the same time, for problems with specific structure, the worst-case guarantees for convergence of gradient methods obtained considering only the the largest and smallest eigenvalue of the Hessian can be too pessimistic. Indeed, we see that in practice, the distribution of eigenvalues can be quite specific, with a relatively small amount of top eigenvalues that are much larger than the others (Fig. 1). Consequently, any a priori information on the structure of the Hessian can be significant from the optimization perspective. Ultimately, we want to have algorithms that are able to benefit from this knowledge, achieving faster rates when the distribution of eigenvalues is far from uniform.

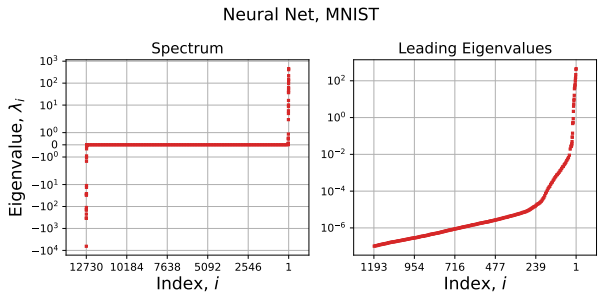


*Figure 2.* Left: spectrum of the Hessians for a two-layer fully connected neural network trained on MNIST dataset. Right: zoomed top eigenvalues. The total number of parameters is $n = 12730$ (the first layer: 12560, the second layer: 170). We see that the dimension $\tau$ of the subspace with positive eigenvalues is much bigger than the dimension of the last layer. However, there are only a few eigenvalues that are significantly larger than the others.

**Contributions.** In this work, we develop spectral preconditioning for the gradient methods that is able to tackle non-convex problems with *highly non-uniform* and clustered spectrum of the Hessian, that we often observe in practice. For that, we propose to make a step back from the common dichotomy between convex and non-convex problems. We introduce a new notion of *graded non-convex* functions, which granulate the class of all non-convex problems into nested family of subclasses. We say, for some integer $\tau \geq 1$

$$f \text{ is non-convex of grade } \tau \quad \Leftrightarrow \quad \nabla^2_\tau f \succeq \mathbf{0}, \tag{2}$$

where $\nabla^2_\tau f$ is composed by the top $\tau$ eigenvectors of the full Hessian (see (6) for the formal definition). For $\tau = n$, where $n$ is dimension of the problem, we obtain the entire Hessian $\nabla^2 f$ and inequality (2) means the standard convexity (1). It appears that for many practical non-convex problems, this condition is satisfied at least for some small $\tau$. For example, for deep neural networks with convex loss it is satisfied at least for $\tau \geq d$, where $d$ is the dimension of the last layer (however, in practice we observe that the actual values of $\tau$ can be much bigger, see Fig. 2).

Inequality (2) provides us with a certain *convex subspace* at each point (such that our function is locally convex in this subspace, see Fig. 3), which looks very attractive from the optimization standpoint. Furthermore, it contains the leading eigenvalues, which constitute the primary computational burden for first-order methods.

Based on our problem class, we propose to use a positive definite matrix $\mathbf{H} \approx \nabla^2_\tau f(\boldsymbol{x})$ as a natural preconditioner for our method, that we call *spectral preconditioning*:

$$\boldsymbol{x}^+ \quad = \quad \boldsymbol{x} - \left(\mathbf{H} + \alpha \mathbf{I}\right)^{-1} \nabla f(\boldsymbol{x}), \tag{3}$$

where and $\alpha \geq 0$ is a regularization parameter. For these iterations, we establish strong convergence guarantees, that improve with increasing the parameter $\tau$ (see Table 1).

Therefore, the method with any $\tau \geq 1$ works *provably better* than the basic gradient method ($\tau = 0$) in terms of the rate of convergence. For $\tau = n$ (the full Hessian), our algorithm becomes the regularized Newton method with the best global complexity bounds known in the literature. However, the most efficient version of our method corresponds to the case of small $\tau \approx 10^0 - 10^2$, when estimating the matrix $\mathbf{H}$ can be done efficiently with the hot-start *power method*.

**Related Work.** Preconditioning is an important tool in numerical analysis and optimization (Nocedal & Wright,

2006). The basic example is preconditioning of the conjugate gradient method (Hestenes & Stiefel, 1952) for solving a system of linear equations. The choice of the right preconditioner is a difficult task and it often depends on an a-priori knowledge on the problem structure, as, for example, the Laplacian preconditioning for the graph-induced problems (Spielman & Teng, 2004; Vaidya, 1991) or for the systems involving partial differential equations (Mardal & Winther, 2011).

In optimization, a powerful approach for preconditioning that works for general problems is called the Newton method (Polyak, 2007). It uses the Hessian matrix $\nabla^2 f$ to alleviate the impact of ill-conditioned characteristics of the Hessian spectrum. The modern versions of this method provide us with the global rates of convergence that are significantly better than those of the first-order algorithms (Nesterov & Polyak, 2006; Cartis et al., 2011a;b; Grapiglia & Nesterov, 2017; Karimireddy et al., 2018; Doikov et al., 2022). However, computing and inverting the Hessian matrix is prohibitively expensive in terms of the arithmetic operations and memory usage to store big matrices.

The common approximate technique for improving the arithmetic cost of the methods with the full Hessian is called the *quasi-Newton methods*, such as SR1, DFP, BFGS, L-BFGS and others (Nocedal & Wright, 2006). Despite these methods show an outstanding performance on practical problems of a moderate dimension, it is a significant challenge to establish a rigorous theory of convergence for *quasi-second-order methods*, which would provably benefit globally from inexact Hessian information, while the local non-asymptotic theory of convergence for the classical quasi-Newton methods has emerged only very recently (Rodomanov & Nesterov, 2021a;b; Jin & Mokhtari, 2023). Note that employing a naive and straightforward line-search or damping approach in the basic Newton method can have as slow convergence as the plain gradient descent (Cartis et al., 2013). See also (Hanzely et al., 2022; Kamzolov et al., 2023; Jiang & Mokhtari, 2023) for the recent advancement in the global analysis of the damped and quasi-Newton methods for convex objectives. Some of the modern scalable techniques for the Newton method are *block-coordinate updates*, *stochastic subspaces* and *sketching* (Hanzely et al., 2020; Nutini et al., 2022; Hanzely, 2023), *distributed* and *lazy* computations (Qian et al., 2021; Safaryan et al., 2021; Islamov et al., 2021; 2022; Doikov et al., 2023; Doikov & Grapiglia, 2023) and *stochastic preconditioning* (Pasechnyuk et al., 2022; 2023), for convex and non-convex problems.

Our analysis is based on the new concept of *graded non-convexity*, which is related to studies of other generalized notions of convexity (Vial, 1983; Hörmander, 2007). At the same time, more refined specifications of the problem class that investigate the distribution of the eigenvalues were considered recently in (Kovalev et al., 2018; Scieur & Pedregosa, 2020; Cunha et al., 2022; Goujaud et al., 2022). The motivation of our work to cut large gaps between the leading eigenvalues is closely related to recently proposed coordinate methods with *volume sampling* (Rodomanov & Kropotov, 2020) and *polynomial preconditioning* technique (Doikov & Rodomanov, 2023), analysing the convex problems with a specific structure. Another similar and relevant work is (Kunstner et al., 2024), which proposes a backtracking procedure for constructing a provable preconditioning for convex problems.

Additionally, several recent works study efficient preconditioning of gradient methods for specific classes of *non-convex* problems using a similar idea of employing the spectrum of the objective Hessian. These include over-parameterized non-convex *matrix factorization* (Zhang et al., 2021; 2023; Ma et al., 2023) and *kernel ridge regression* problems (Ma & Belkin, 2017). Compared to these works, our preconditioning is designed to solve general smooth and possibly non-convex problems without assumptions on the intrinsic structure of the objective, but rather on the distribution of the Hessian spectrum.

**Contents.** The rest of the paper is organized as follows. In Section 2 we introduce the notion of *graded non-convexity*. We study its main properties and provide several examples. Section 3 contains our main algorithm (14) the Gradient Method with Spectral Preconditioning. We prove fast convergence rates for this method, showing an improvement when increasing the preconditioning order (Theorem 4.2). In Section 5 we show a simple modification of our method that allows to remove the dependency on the negative part of the spectrum (Theorem 5.1). In Section 6 we show improved rates of convergence for the special case of convex functions. In Section 7 we discuss the efficient implementation of our algorithms. Section 8 presents numerical experiments. Missing proofs are provided in the appendix.

**Notation.** We are interested in solving the following unconstrained optimization problem[1]

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} f(\boldsymbol{x}) \tag{4}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a several times differentiable target function, which can be *non-convex*. We denote its global lower bound by $f^\star := \inf_{\boldsymbol{x}} f(\boldsymbol{x})$, which we assume to be finite. In non-convex optimization, our goal is to find an approximate *stationary point* $\bar{\boldsymbol{x}}$ to (4), ensuring $\|\nabla f(\bar{\boldsymbol{x}})\| \leq \varepsilon$, for some $\varepsilon > 0$. We use the standard Euclidean norm for vectors: $\|\boldsymbol{x}\| := \langle \boldsymbol{x}, \boldsymbol{x} \rangle^{1/2}$, $\boldsymbol{x} \in \mathbb{R}^n$, and the corresponding operator norm for matrices: $\|\mathbf{A}\| := \max_{\boldsymbol{x}: \|\boldsymbol{x}\| \leq 1} \|\mathbf{A}\boldsymbol{x}\|$.

[1] Our results can be also generalized to the *composite* formulation of optimization problems, see Section C in the appendix.

We denote the gradient of $f$ at point $\boldsymbol{x} \in \mathbb{R}^n$ by $\nabla f(\boldsymbol{x}) \in \mathbb{R}^n$, and the Hessian matrix by $\nabla^2 f(\boldsymbol{x}) \in \mathbb{R}^{n \times n}$. Note that the Hessian is a symmetric matrix. Hence, for any point $\boldsymbol{x} \in \mathbb{R}^n$ we can introduce the following spectral decomposition:

$$\nabla^2 f(\boldsymbol{x}) \quad = \quad \sum_{i=1}^{n} \lambda_i(\boldsymbol{x}) \boldsymbol{u}_i(\boldsymbol{x}) \boldsymbol{u}_i(\boldsymbol{x})^\top, \tag{5}$$

where the eigenvalues are sorted in a *non-ascending* order: $\lambda_1(\boldsymbol{x}) \geq \lambda_2(\boldsymbol{x}) \geq \ldots \geq \lambda_n(\boldsymbol{x})$, and $\boldsymbol{u}_1(\boldsymbol{x}), \ldots, \boldsymbol{u}_n(\boldsymbol{x}) \in \mathbb{R}^n$ are the corresponding orthonormal eigenvectors. There are always several possible choices for the eigenvector basis, hence, we assume that a specific selection has been fixed. Our results remain independent of the particular selection. For a fixed spectral decomposition (5), we denote by $\nabla^2_\tau f(\boldsymbol{x})$, $1 \leq \tau \leq n$, the *Hessian of spectral order* $\tau$:

$$\nabla^2_\tau f(\boldsymbol{x}) \quad := \quad \sum_{i=1}^{\tau} \lambda_i(\boldsymbol{x}) \boldsymbol{u}_i(\boldsymbol{x}) \boldsymbol{u}_i(\boldsymbol{x})^\top \in \mathbb{R}^{n \times n}. \tag{6}$$

For convenience, we set $\nabla^2_0 f(\boldsymbol{x}) \equiv \mathbf{0}$. Thus, $\operatorname{rank}(\nabla^2_\tau f(\boldsymbol{x})) \leq \tau$, and for $\tau = n$ we obtain the full Hessian. Of course, the decomposition of the form (6) is not unique, especially if $\lambda_\tau(\boldsymbol{x}) = \lambda_{\tau+1}(\boldsymbol{x})$ for a certain $\boldsymbol{x}$. However, we always assume that a unique choice has been fixed for ease of notation. We denote by $\nabla^3 f(\boldsymbol{x})$ the third derivative of $f$ at point $\boldsymbol{x}$, which is a trilinear symmetric form. The action of this form onto some fixed directions $\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3 \in \mathbb{R}^n$ is denoted by $\nabla^3 f(\boldsymbol{x})[\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3] \in \mathbb{R}$.

## 2. Problem Classes

A standard assumption in non-convex optimization is to assume that the objective function is smooth, i.e. that its gradients are Lipschitz continuous. For twice-differentiable functions, this is equivalent to assuming that the norm of the Hessian, $\|\nabla^2 f(\boldsymbol{x})\|$, is uniformly bounded. In this section, we introduce a new concept that allows us to capture the distribution of the eigenvalues in more detail.

### 2.1. Grade of Non-convexity

We start with a formal definition of our problem class.

**Definition 2.1.** For a twice continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ and convex set $Q \subseteq \mathbb{R}^n$, we say that $f$ is *non-convex of grade* $\tau$, if

$$\boxed{\nabla^2_\tau f(\boldsymbol{x}) \quad \succeq \quad \mathbf{0}, \qquad \forall \boldsymbol{x} \in Q.} \tag{7}$$

In other words, (7) means that the top $\tau$ eigenvalues of the Hessian are non-negative everywhere on $Q$:

$$\lambda_\tau(\boldsymbol{x}) \quad \geq \quad 0, \qquad \forall \boldsymbol{x} \in Q. \tag{8}$$

In our analysis, we will mostly use for simplicity $Q \equiv \mathbb{R}^n$ (the whole space). However, it can only refine our problem class if some localization of a solution $\boldsymbol{x}^\star \in Q$ is available (see also our discussion in Section 9). Definition 2.1 implies a certain restriction on a surface structure of the objective function. In differential geometry, condition (8) on the curvatures leads to the notion of $\tau$-*convex surface* (Gromov, 1991).

For $0 \leq \tau \leq n$, we denote by $\mathcal{F}_\tau$ the set of functions $f \in \mathcal{F}_\tau$ that satisfies (7). By our definition, $\mathcal{F}_0 = C^2(\mathbb{R}^n)$ is the set of *all* twice continuously differentiable functions[2] and $\mathcal{F}_n$ consists only of convex objectives. These classes are closed under multiplications by a non-negative scalar: $f \in \mathcal{F}_\tau \Rightarrow \alpha f \in \mathcal{F}_\tau$ for any $\alpha \geq 0$. Hence, we obtain the *nested family of functional cones*:

$$\underset{\text{all functions}}{\mathcal{F}_0} \quad \supset \quad \mathcal{F}_1 \quad \supset \quad \ldots \quad \supset \quad \mathcal{F}_{n-1} \quad \supset \quad \underset{\text{convex functions}}{\mathcal{F}_n}$$

Intuitively, functions with larger grades should be easier to minimize. At the same time, a method that works for a certain $\mathcal{F}_\tau$ can also tackle all problems from $\mathcal{F}_i$ for $i \geq \tau$.

### 2.2. Main Properties

Let us study some of the most basic properties that follow from our definition. First, we have the following important *grading rule*, that equips our sets with additional structure.

**Proposition 2.2.** *Let $f \in \mathcal{F}_i$ and $g \in \mathcal{F}_j$, for some $0 \leq i, j \leq n$ such that $i + j \geq n$. Then, it holds:*

$$\begin{aligned} f + g \quad &\in \quad \mathcal{F}_{i+j-n}, \\ \operatorname{smax}(f, g) \quad &\in \quad \mathcal{F}_{i+j-n}, \end{aligned}$$

*where* $\operatorname{smax}(f, g)(\boldsymbol{x}) \stackrel{\text{def}}{=} \ln(e^{f(\boldsymbol{x})} + e^{g(\boldsymbol{x})})$ *is the soft maximum of two functions.*

In particular, the summation of a function $f \in \mathcal{F}_\tau$ with *any convex* function $g \in \mathcal{F}_n$ cannot decrease its grade $\tau$. The same holds for (soft) maximum operations[3]. Next, we observe that the grade is preserved under affine substitutions.

**Proposition 2.3.** *Let $f \in \mathcal{F}_\tau(\mathbb{R}^n)$ be non-convex of grade $\tau$, and let $\mathbf{A} \in \mathbb{R}^{n \times m}, \boldsymbol{b} \in \mathbb{R}^n$, with $m + \tau \geq n$. Denote $g(\boldsymbol{x}) = f(\mathbf{A}\boldsymbol{x} + \boldsymbol{b})$. Then, $g \in \mathcal{F}_{m-n+\tau}(\mathbb{R}^m)$ is non-convex of grade $m - n + \tau$.*

For $\tau \geq 1$, our Definition 2.1 implies that the Hessian is not negative definite at any point: $\nabla^2 f(\boldsymbol{x}) \not\succeq \mathbf{0}$. This condition means that the function *cannot have strict local maxima*. This fact can be formalized as follows.

[2] Note that definition of our classes $\mathcal{F}_\tau, 0 \leq \tau \leq n$ depends on set $Q$ and on the problem dimension $n$. Thus, it would be more formal to use notation $\mathcal{F}^n_\tau(Q)$. However, we omit extra indices since they should always be clear from the context.

[3] It holds $\max(f, g) = \lim_{\mu \to 0} \mu \cdot \operatorname{smax}(f/\mu, g/\mu)$. We prefer to work with soft max operation to keep all functions in the smooth class.

**Proposition 2.4.** *Let $f \in \mathcal{F}_\tau(Q)$ for $\tau \geq 1$. Then, the weak maximum principle holds: for any compact $K \subset Q$, the maximum is always achieved at the boundary,*

$$\max_{\boldsymbol{x} \in K} f(\boldsymbol{x}) \quad = \quad \max_{\boldsymbol{x} \in \partial K} f(\boldsymbol{x}).$$

Finally, let us mention an intuitive geometric description of the surface of function $f$ of a certain grade $\tau$.

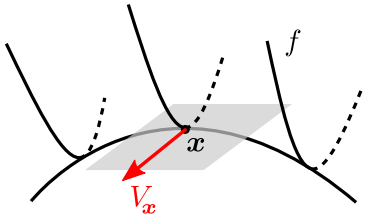


*Figure 3.* A surface of a non-convex function $f$. At point $\boldsymbol{x}$ there is a subspace $V_{\boldsymbol{x}}$ where $f$ is convex.

**Proposition 2.5.** *Let for any $\boldsymbol{x}$ there exists a vector subspace $V_{\boldsymbol{x}} \subseteq \mathbb{R}^n$ with $\dim(V_{\boldsymbol{x}}) \geq \tau$ such that*

$$f(\boldsymbol{x} + \boldsymbol{h}) \quad \geq \quad f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{h} \rangle, \quad \forall \boldsymbol{h} \in V_{\boldsymbol{x}}. \qquad (9)$$

*Then $f \in \mathcal{F}_\tau$ is non-convex of grade $\tau$.*

This inequality is stronger than our Definition 2.1: (9) is a sufficient condition for (7). Geometrically it means that for every point $\boldsymbol{x}$ there exists a subspace $V_{\boldsymbol{x}}$ of the tangent space to the surface such that restriction of $f$ onto this subspace is convex (see Fig. 3).

### 2.3. Examples

In this section, we provide several examples of non-convex objective functions of a non-trivial grade of non-convexity.

**Example 2.6** (Quadratic Functions)**.** *Let $f(\boldsymbol{x}) = \frac{1}{2}\langle \mathbf{A}\boldsymbol{x}, \boldsymbol{x} \rangle - \langle \boldsymbol{b}, \boldsymbol{x} \rangle$ for some $\mathbf{A} = \mathbf{A}^\top \in \mathbb{R}^{n \times n}$ and $\boldsymbol{b} \in \mathbb{R}^n$. Let the top $\tau$ eigenvalues of $\mathbf{A}$ be positive: $\lambda_1(\mathbf{A}) \geq \ldots \geq \lambda_\tau(\mathbf{A}) \geq 0$. Then $f \in \mathcal{F}_\tau$.*

Consequently, adding a power of the Euclidean norm as a simple regularizer to a non-convex quadratic function as in Example 2.6, we obtain for $p > 2$ and some $\sigma > 0$:

$$f(\boldsymbol{x}) \quad = \quad \tfrac{1}{2}\langle \mathbf{A}\boldsymbol{x}, \boldsymbol{x} \rangle - \langle \boldsymbol{b}, \boldsymbol{x} \rangle + \tfrac{\sigma}{p}\|\boldsymbol{x}\|^p, \qquad (10)$$

which describes a simple family of non-convex objectives that can realize all possible grades $f \in \mathcal{F}_\tau$, $0 \leq \tau \leq n$, while a global solution $\boldsymbol{x}^\star = \operatorname{argmin}_{\boldsymbol{x}} f(\boldsymbol{x})$ always exists, since $f$ has bounded sublevel sets. The problems of the form (10) are important in applications to regularized second-order and high-order methods (Grapiglia & Nesterov, 2017; Nesterov, 2019; 2022).

**Example 2.7** (Low-rank Vector Fields)**.** *Let $\varphi : \mathbb{R} \to \mathbb{R}$ be a univariate (possibly non-convex) function, and $\boldsymbol{u} : \mathbb{R}^n \to \mathbb{R}^n$ be a general differentiable mapping. Set*

$$f(\boldsymbol{x}) \quad = \quad \varphi(\langle \boldsymbol{u}(\boldsymbol{x}), \boldsymbol{x} \rangle), \qquad \boldsymbol{x} \in \mathbb{R}^n. \qquad (11)$$

*Assume that there exists $r \leq n-1$ such that for any $\boldsymbol{x}$, it holds $r \geq \operatorname{rank}\big(\nabla \boldsymbol{u}(\boldsymbol{x}) + \nabla^2 \boldsymbol{u}(\boldsymbol{x})\boldsymbol{x}\big)$. Then, $f : \mathbb{R}^n \to \mathbb{R}$ is non-convex of grade $n - r - 1$.*

A particular case is when $\boldsymbol{u}(\boldsymbol{x}) \equiv \boldsymbol{u} \in \mathbb{R}^n$ is a constant vector field. Then, we obtain that the function (11) is non-convex of grade $n-1$ (see Fig. 4).

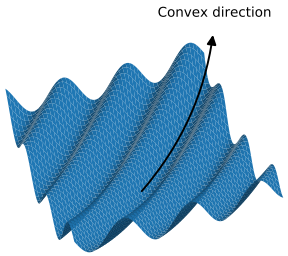


*Figure 4.* The graph of two-dimensional function $f(x, y) = \sin(x + y) + q(x, y)$, where $q$ is a convex quadratic. The non-convex component has the structure of (11) with $\boldsymbol{u} \equiv (1, 1)^\top$.

**Example 2.8** (Partial Convexity)**.** *Let $f(\boldsymbol{x}, \boldsymbol{y})$ depend on two groups of variables $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{y} \in \mathbb{R}^m$. Assume that for any fixed $\boldsymbol{y}$, the function of the first argument*

$$f(\cdot, \boldsymbol{y}) : \ \mathbb{R}^n \to \mathbb{R}$$

*is convex. Then $f : \mathbb{R}^{n+m} \to \mathbb{R}$ is non-convex of grade $n$.*

We see that if the function is convex with respect to *some* of the variables, then, as a function of all variables, it is non-convex of the corresponding grade. Note, however, that the actual structure of the subspace when the Hessian is positive can be quite complicated. As a direct consequence, we obtain that matrix factorizations and deep neural network models with convex losses satisfy our assumption.

**Example 2.9** (Diagonal Neural Networks)**.** *For a given $\boldsymbol{c} \in \mathbb{R}^n$, consider*

$$f(\boldsymbol{x}, \boldsymbol{y}) \quad = \quad \tfrac{1}{2}\|\boldsymbol{x} \circ \boldsymbol{y} - \boldsymbol{c}\|^2, \quad \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n, \qquad (12)$$

*where $\circ$ is the element-wise product. At every point $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{2n}$ the Hessian has the following set of $2n$ eigenvalues, a pair for each $1 \leq i \leq n$: $\lambda_i = \frac{1}{2}\big[x_i^2 + y_i^2 \pm \sqrt{(x_i^2 - y_i^2)^2 + 4(2x_i y_i - c_i)^2}\,\big]$, with at least one being always non-negative. Therefore, $f : \mathbb{R}^{2n} \to \mathbb{R}$ is non-convex of grade $n$.*

The objective of the form (12) is a good model for studying the dynamics of gradient methods in deep learning (Woodworth et al., 2020; Pesme et al., 2021; Even et al., 2023; Pesme & Flammarion, 2023).

**Example 2.10** (Matrix Factorizations)**.** *For a given target matrix* $\mathbf{C} \in \mathbb{R}^{n\times m}$*, and some* $r > 0$*, consider*

$$f(\mathbf{X}, \mathbf{Y}) \quad = \quad \tfrac{1}{2}\|\mathbf{X}\mathbf{Y} - \mathbf{C}\|_F^2, \tag{13}$$

*where* $\mathbf{X} \in \mathbb{R}^{n\times r}$, $\mathbf{Y} \in \mathbb{R}^{r\times m}$*. Then,* $f : \mathbb{R}^{n\times r+r\times m} \to \mathbb{R}$ *is non-convex of grade* $\max\{m, n\} \times r$*. More generally, the function* $f(\mathbf{X}_1, \ldots, \mathbf{X}_d) = \frac{1}{2}\|\mathbf{X}_1\mathbf{X}_2 \cdots \mathbf{X}_d - \mathbf{C}\|_F^2$*,* $\mathbf{X}_i \in \mathbb{R}^{n_i\times m_i}$*, is non-convex of grade* $\max_{1\le i\le d}\big[n_i \times m_i\big]$.

We observe that in many practical scenarios, it is very common to encounter objectives that have a subspace with positive eigenvalues of the Hessian. Indeed, the opposite seems rather quite rare and indicates that the target objective is *purely concave*. On the contrary, for deep neural network models with convex losses we always ensure the existence of a subspace with *positive curvature*, which serves as the main computational burden for a method to converge to a stationary point, when the problem is ill-conditioned.

In the next sections, we propose the spectral preconditioning technique for gradient methods in order to tackle ill-conditioning of the positive part of the spectrum. At the same time, as we will show, the impact of the negative curvature to optimization methods can easily be alleviated.

## 3. Spectral Preconditioning

In this section we present our proposed algorithm. The method aims to exploit the (possibly) convex-like structure of the function. When there is no such structure ($\tau = 0$) the method becomes equal to the standard gradient method.

At every iteration of our method, we choose a matrix $\mathbf{H} = \mathbf{H}^\top \succeq 0$ and perform the following preconditioned gradient step, for a given point $\boldsymbol{x} \in \mathbb{R}^n$ and gradient vector $\boldsymbol{g} \in \mathbb{R}^n$:

$$\text{GradStep}_{\mathbf{H},\alpha}(\boldsymbol{x}, \boldsymbol{g}) \quad := \quad \boldsymbol{x} - \big(\mathbf{H} + \alpha\mathbf{I}\big)^{-1}\mathbf{g},$$

where $\alpha \ge 0$ is some regularization parameter. Hence, the matrix $\mathbf{H}$ plays the role of a *preconditioner*. We want to choose it as an approximation of the Hessian of a certain spectral order: $\mathbf{H} \approx \nabla^2_\tau f(\boldsymbol{x})$. When $\tau = 0$, we have $\nabla^2_\tau f(\boldsymbol{x}) \equiv \mathbf{0}$ and thus we do a step of plain gradient descent. In contrast, for $\tau = n$ we approximate the full Newton step. Let us present the method in algorithmic form.

**Gradient Method with Spectral Preconditioning** (14)

**Choose** $\boldsymbol{x}_0 \in \mathbb{R}^n$ and $0 \le \tau \le n$.

**For** $k \ge 0$ **iterate:**

1. Estimate $\mathbf{H}_k \approx \nabla^2_\tau f(\boldsymbol{x}_k) \in \mathbb{R}^{n\times n}$
2. Perform the gradient step, for some $\alpha_k \ge 0$:

$$\boldsymbol{x}_{k+1} \quad = \quad \text{GradStep}_{\mathbf{H}_k,\alpha_k}(\boldsymbol{x}_k, \nabla f(\boldsymbol{x}_k))$$

Up to now, we do not specify explicitly how we estimate $\nabla^2_\tau f(\boldsymbol{x}_k)$. Our matrix $\mathbf{H}_k$ should be easily computable and we aim to maintain a low rank representation,

$$\mathbf{H}_k \quad = \quad \sum_{i=1}^{\tau} a_{k,i}\boldsymbol{v}_{k,i}\boldsymbol{v}_{k,i}^\top, \tag{15}$$

for a set of positive numbers $(a_{k,i})_{i=1}^\tau$ and orthonormal vectors $(\boldsymbol{v}_{k,i})_{i=1}^\tau$, so that we can perform iterations of algorithm (14) cheaply. We discuss details on efficient implementation of every step in Section 7. To quantify the approximation errors, we denote

$$\delta_k \quad := \quad \|\mathbf{H}_k - \nabla^2_\tau f(\boldsymbol{x}_k)\|, \quad \delta \; := \; \max_k \delta_k. \tag{16}$$

Thus, if $\delta = 0$, we use the exact Hessian of spectral order $\tau$.

## 4. Global Convergence

Let us show the main convergence result for algorithm (14). We establish fast convergence rates to a stationary point of our objective (4), starting from an arbitrary initial point $\boldsymbol{x}_0$. These rates become better when increasing the spectral order parameter $\tau$ used in our method.

We introduce the following characteristics of smoothness for our problem classes. We assume that the Hessian is Lipschitz continuous, with parameter $L \ge 0$, for all $\boldsymbol{x}, \boldsymbol{y}$:

$$\|\nabla^2 f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{y})\| \quad \le \quad L\|\boldsymbol{x} - \boldsymbol{y}\|. \tag{17}$$

To estimate the error of using the Hessian of a spectral order $\tau$, we also introduce the following system of parameters:

$$\sigma_\tau \quad := \quad \sup_{\boldsymbol{x}\in\mathbb{R}^n} \|\nabla^2 f(\boldsymbol{x}) - \nabla^2_\tau f(\boldsymbol{x})\|. \tag{18}$$

Thus, $\sigma_0 := \sup_{\boldsymbol{x}} \|\nabla^2 f(\boldsymbol{x})\|$ is simply the (best) Lipschitz constant of the gradient of $f$, and $\sigma_n = 0$. In general, the value of $\sigma_\tau$ is the uniform bound for both the $(\tau+1)th$ eigenvalue of the Hessian, and its negative part, for $\tau < n$:

$$\sigma_\tau \quad \ge \quad \max\{\lambda_{\tau+1}(\boldsymbol{x}), -\lambda_n(\boldsymbol{x})\}, \quad \forall \boldsymbol{x} \in \mathbb{R}^n. \tag{19}$$

Under these conditions, and using a sufficiently big regularization parameter $\alpha_k$ (a second-order "step size"), we can prove the following progress for one iteration $\boldsymbol{x}_k \mapsto \boldsymbol{x}_{k+1}$ of our method.

**Lemma 4.1.** *Let* $\alpha_k \ge \sqrt{\frac{L\|\nabla f(\boldsymbol{x}_k)\|}{2}} + \sigma_\tau + \delta$*. Then*

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k+1}) \quad \ge \quad \tfrac{1}{8\alpha_k}\|\nabla f(\boldsymbol{x}_{k+1})\|^2. \tag{20}$$

Note that we do not need to know the exact values of the parameters $L$, $\sigma_\tau$, and $\delta_k$. In practice, we can use an adaptive search to ensure sufficient progress (20). This lemma leads us to the basic global convergence result.

**Theorem 4.2.** *Let $f \in \mathcal{F}_\tau$ be non-convex of grade $\tau$, where $0 \le \tau \le n$ is fixed. Let $f$ have a Lipschitz Hessian with constant $L$ and bounded parameter $\sigma_\tau$. Consider iterations $\{\boldsymbol{x}_k\}_{k\ge 0}$ of algorithm (14) with*

$$\alpha_k \;=\; \sqrt{\tfrac{L\|\nabla f(\boldsymbol{x}_k)\|}{2}} + \sigma_\tau + \delta. \qquad (21)$$

*Then, for any $\varepsilon > 0$, it is enough to do $K =$*

$$\left\lceil 8(f(\boldsymbol{x}_0) - f^\star) \cdot \left(\sqrt{\tfrac{L}{2}}\tfrac{1}{\varepsilon^{3/2}} + \tfrac{\sigma_\tau+\delta}{\varepsilon^2}\right) \;+\; 2\ln\tfrac{\|\nabla f(\boldsymbol{x}_0)\|}{\varepsilon} \right\rceil$$

*steps to ensure $\min\limits_{1\le i\le K} \|\nabla f(\boldsymbol{x}_i)\| \le \varepsilon$.*

For $\tau = n$ (the full Newton), we have $\sigma_n = 0$ and the complexity becomes $\mathcal{O}(\frac{1}{\varepsilon^{3/2}})$ up to an additive logarithmic term. It can also be established for the cubic regularization of the Newton method (Nesterov & Polyak, 2006). For $\tau = 0$ we recover the rate of the gradient descent on non-convex problems (Nesterov, 2018). Note that the rule (21) for choosing $\alpha_k$ is very simple. It is inspired by the gradient regularization technique developed initially for convex optimization (Mishchenko, 2023; Doikov & Nesterov, 2023).

## 5. Cutting the Negative Spectrum

In our previous result, we saw that the complexity of the method depends on the parameter $\sigma_\tau$, which becomes smaller when increasing the spectral order $\tau$ of the method. However, due to (19), it includes a bound for the absolute value of the negative curvature, which can be constantly big. In this section, we propose a simple modification of our step-size rule, which alleviates this issue.

Let us denote the *positive part* of the Hessian, $\nabla^2_+ f(\boldsymbol{x}) := \nabla^2_\xi f(\boldsymbol{x})$, where $\xi = \operatorname{argmax}\{\tau : \nabla^2_\tau f(\boldsymbol{x}) \succeq \mathbf{0}\}$. Correspondingly, the *negative part* is $\nabla^2_- f(\boldsymbol{x}) := \nabla^2_+ f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{x}) \succeq \mathbf{0}$. We introduce the following system of parameters (compare with (18)):

$$\sigma^+_\tau \;:=\; \sup_{\boldsymbol{x}\in\mathbb{R}^n} \|\nabla^2_+ f(\boldsymbol{x}) - \nabla^2_\tau f(\boldsymbol{x})\|. \qquad (22)$$

So, $\sigma^+_\tau$ is the uniform bound for the positive tail of the spectrum, and it is no longer affected by the negative curvature:

$$\sigma^+_\tau \;\ge\; \max\{\lambda_{\tau+1}(\boldsymbol{x}),\, 0\}, \quad \forall \boldsymbol{x} \in \mathbb{R}^n. \qquad (23)$$

Our new step-size rule is based on the cubic regularization technique. Namely, at every iteration $k \ge 0$, we compute the regularization parameter as the maximum of the following univariate concave function:

$$\alpha^\star_k = \operatorname*{argmax}_{\alpha>0} \left[ -\tfrac{1}{2}\langle (\mathbf{H}_k + (\alpha+\eta)\mathbf{I})^{-1}\boldsymbol{g}_k, \boldsymbol{g}_k\rangle - \tfrac{2\alpha^3}{3L^2} \right],$$

where $\boldsymbol{g}_k := \nabla f(\boldsymbol{x}_k)$ is the current gradient, $\eta \ge 0$ is a balancing term, and $L > 0$ is the Lipschitz constant of the Hessian. This subproblem is well defined and has a unique global maximum, which can be found by using binary search or univariate Newton iterations (Conn et al., 2000; Nesterov & Polyak, 2006). Then, to eliminate the negative curvature, for the sequence generated by our method we use an extra sequence of test points $\{\boldsymbol{y}_k\}_{k\ge 1}$ defined by $\boldsymbol{y}_{k+1} := \boldsymbol{x}_k + \mathbf{P}_k(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k)$, where $\mathbf{P}_k$ is a projector which preserves the image of $\mathbf{H}_k$: $\mathbf{H}_k\mathbf{P}_k = \mathbf{H}_k$, but has a small intersection with the negative part of the Hessian,

$$\|\nabla^2_- f(\boldsymbol{x}_k)\mathbf{P}_k\| = \|(\nabla^2 f(\boldsymbol{x}_k) - \nabla^2_+ f(\boldsymbol{x}_k))\mathbf{P}_k\| \;\le\; \delta_-,$$

for some $\delta_- > 0$. It can be built directly from the low-rank representation (15) of $\mathbf{H}_k$, as follows: $\mathbf{P}_k = \sum_{i=1}^\tau \boldsymbol{v}_{k,i}\boldsymbol{v}_{k,i}^\top$. Therefore, this matrix comes with no extra cost. We have $\delta_- = 0$ when $\boldsymbol{v}_{k,i}$ are orthogonal to all eigenvectors of $\nabla^2_- f(\boldsymbol{x}_k)$. Note that the tolerance parameters $\delta$ and $\delta_-$ are induced by the approximation error $\mathbf{H}_k \approx \nabla^2_\tau f(\boldsymbol{x}_k)$, and, contrary to $\sigma_\tau$ and $\sigma^+_\tau$, they do not describe our problem class. Using a high accuracy approximations for matrices $\mathbf{H}_k$ (see Section 7), we can make both $\delta$ and $\delta_-$ arbitrarily close to zero. We are ready to state the better complexity result for algorithm (14) with a convergence rate that is independent of the negative curvature.

**Theorem 5.1.** *Let $f \in \mathcal{F}_\tau$ be non-convex of grade $\tau$, where $0 \le \tau \le n$ is fixed. Let $f$ have a Lipschitz Hessian with constant $L$ and bounded parameter $\sigma^+_\tau$. Consider iterations $\{\boldsymbol{x}_k\}_{k\ge 0}$ of algorithm (14) with*

$$\alpha_k \;=\; \alpha^\star_k + \eta, \qquad (24)$$

*where $\eta := \sigma^+_\tau + \delta + \delta_-$ is the balancing term. Then, for any $\varepsilon > 0$, it is enough to do*

$$K \;=\; \left\lceil 2(f(\boldsymbol{x}_0) - f^\star)\left(\tfrac{3\sqrt{2L}}{\varepsilon^{3/2}} + \tfrac{16\eta}{\varepsilon^2}\right)\right\rceil$$

*steps to ensure $\min\limits_{1\le i\le K} \|\nabla f(\boldsymbol{y}_i)\| \le \varepsilon$.*

This complexity bound is similar to that one of Theorem 4.2, where we substituted $\sigma_\tau \mapsto \sigma^+_\tau$. It can be much better in case of large negative eigenvalues of the Hessian, since then $\sigma^+_\tau$ is much smaller than $\sigma_\tau$. The cost of such an improvement is a more complex rule (24), involving $\alpha^\star_k$, and the guarantee is given for the auxiliary points $\{\boldsymbol{y}_k\}_{k\ge 1}$.

## 6. Convex Convergence Analysis

In this section, we provide the analysis of our method for a specific case of $f \in \mathcal{F}_n$ (convex objectives). In this case, we do not have the negative curvature part: $\nabla^2_- f(\boldsymbol{x}) \equiv \mathbf{0}$, and $\sigma^+_\tau \equiv \sigma_\tau$. Since $\mathcal{F}_n \subset \mathcal{F}_\tau$ for any $\tau$, the results of the previous sections can be applied directly for the convex case. However, as we show, using a more refined technique, we can prove much better convergence rates for convex

and strongly convex problems. We denote by $\mu \geq 0$ the parameter of strong convexity, such that $\nabla^2 f(\boldsymbol{x}) \succeq \mu \mathbf{I}, \forall \boldsymbol{x}$.

When the Hessian is positive semidefinite, we can naturally use it define the *local norm* at point $\boldsymbol{x}$ by $\|\boldsymbol{u}\|_{\boldsymbol{x}} := \langle \nabla^2 f(\boldsymbol{x})\boldsymbol{u}, \boldsymbol{u}\rangle^{1/2}$. The local norm becomes more appropriate for describing the right second-order geometry of the objective (Nesterov & Nemirovski, 1994). With its help, it is possible to characterize smoothness more accurately. Now, we assume that $f$ is *quasi-self-concordant*[4] with parameter $M \geq 0$, for all $\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$:

$$\nabla^3 f(\boldsymbol{x})[\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{v}] \quad \leq \quad M\|\boldsymbol{u}\|_{\boldsymbol{x}}^2 \|\boldsymbol{v}\|. \tag{25}$$

This condition was considered in (Bach, 2010; Karimireddy et al., 2018; Sun & Tran-Dinh, 2019; Doikov, 2023). It appears that many practical problems satisfy this assumption, including *Logistic Regression*, *Soft Maximum*, *Matrix Balancing*, and *Matrix Scaling* problems. We adapt algorithm (14) to this problem class, establishing fast convergence rates that improve with increase of the spectral order $\tau$. We denote by $D := \sup\{\|\boldsymbol{x} - \boldsymbol{x}^\star\| \, : \, f(\boldsymbol{x}) \leq f(\boldsymbol{x}_0)\}$ the diameter of the initial sublevel set which we assume to be bounded. We establish the following result.

**Theorem 6.1.** *Let $f \in \mathcal{F}_n$ be strongly convex with $\mu > 0$ and quasi-self-concordant with constant $M$. Consider iterations $\{\boldsymbol{x}_k\}_{k\geq 0}$ of algorithm (14) with*

$$\alpha_k \quad = \quad M\|\nabla f(\boldsymbol{x}_k)\| + \sigma_\tau + \delta. \tag{26}$$

*Then, for any $\varepsilon > 0$, it is enough to do $K =$*

$$4\left\lceil \left(MD + \tfrac{\sigma_\tau + \delta}{2\mu}\right) \ln \tfrac{f(\boldsymbol{x}_0) - f^\star}{\varepsilon} \; + \; \ln \tfrac{\|\nabla f(\boldsymbol{x}_0)\| D}{\varepsilon} \right\rceil$$

*steps to ensure $f(\boldsymbol{x}_K) - f^\star \leq \varepsilon$.*

*Remark* 6.2. Note that by adding the quadratic regularizer to our objective with small $\mu := \frac{\varepsilon}{D^2}$, we can turn any convex problem into strongly convex one. Hence, we obtain the following complexity for the general convex case:

$$\mathcal{O}\Big(\tfrac{(\sigma_\tau + \delta)D^2}{\varepsilon} + MD \ln \tfrac{f(\boldsymbol{x}_0) - f^\star}{\varepsilon} + \ln \tfrac{\|\nabla f(\boldsymbol{x}_0)\| D}{\varepsilon}\Big).$$

Let us consider the simplest example of using only the one top eigenvector, $\tau = 1$. Then, employing the basic power method (see Section 7 and, e.g. (Golub & Van Loan, 2013)), the complexity of computing the preconditioner is $\mathcal{O}(\frac{\lambda_1}{\lambda_1 - \lambda_2} \ln \frac{1}{\delta})$, where $\delta > 0$ is our target accuracy for computing the approximate eigenvector. Note that $\delta$ enters *under logarithm*. Hence, it seems reasonable to assume that we can make $\delta$ sufficiently small with only a small amount of extra effort. Combining the outer complexity and the inner complexity of the power method, we observe that in some cases, we gain a provable benefit from using the preconditioner. Thus, in the convex case, the total complexity (taking into account the iterations of the power method) becomes $\tilde{\mathcal{O}}\big(\big[\frac{\lambda_2}{\lambda_n} + MD\big] \times \big[\frac{\lambda_1}{\lambda_1 - \lambda_2}\big]\big)$, where $\tilde{\mathcal{O}}(\cdot)$ hides logarithmic factors. It is much better than the complexity $\tilde{\mathcal{O}}(\frac{\lambda_1}{\lambda_n})$ of the pure gradient descent when $\lambda_1 \gg \lambda_2$. These observations can be generalized to an arbitrary $\tau$. However, even for $\tau = 1$, we can observe a significant improvement, which is also justified in our numerical examples (Fig. 11).

[4] Note that for the functions with Lipschitz continuous Hessian (17), we can bound the third derivative with the product of Euclidean norms: $\nabla^3 f(\boldsymbol{x})[\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{v}] \leq L\|\boldsymbol{u}\|^2\|\boldsymbol{v}\|$, which is in most cases less accurate than (25).

## 7. Efficient Implementation

The method (14) relies on an approximation $\mathbf{H}_k \approx \nabla^2_\tau f(\boldsymbol{x}_k)$. We acknowledge that it might be costly to find such an approximation in general. Below, we outline a method that iteratively constructs low rank approximations by employing $T_k \geq 0$ steps of the power method. Here $T_k$ is our parameter. In our experiments we use $T_k = 1$ (combined with hot-start from $\mathbf{H}_{k-1}$). However, other options are to choose $T_k = 0$ for some of the iterations (no update), or to spend more iteration in the initialization phase. We leave the explorations of such schedules to future work.

**Power Method.** For our experiments, we use a generalization of the classical power method for finding the top $\tau$ eigenvectors of a given matrix, which is also known as *orthogonal iterations*. We can write our matrix as $\mathbf{H}_k = \mathbf{V}_k \mathrm{Diag}\,(\boldsymbol{a}_k)\mathbf{V}_k^\top$, where $\boldsymbol{a}_k \in \mathbb{R}^\tau_{>0}$ and $\mathbf{V}_k \in \mathbb{R}^{n\times\tau}$ consists of orthonormal columns $\{\boldsymbol{v}_{k,i}\}_{i=1}^\tau$. We denote by $[\mathbf{A}]_Q := \mathbf{Q}$ the orthogonal matrix from the resulting QR-factorization of $\mathbf{A} = \mathbf{QR}$. It can be implemented as the standard Gram–Schmidt orthogonalization process with arithmetic complexity $\mathcal{O}(\tau^2 n)$ operations. Then, for each $k \geq 0$ we use the following procedure to update matrix $\mathbf{V}_k$:

**Power Method**

Set $\mathbf{V}_k := \mathbf{V}_{k-1}$ if $k \geq 1$ else random init.
**For** $t = 1 \ldots T_k$ **iterate:**
Update $\mathbf{V}_k := \big[\nabla^2 f(\boldsymbol{x}_k)\mathbf{V}_k\big]_Q$
Set $a_{k,i} := \langle \nabla^2 f(\boldsymbol{x}_k)\boldsymbol{v}_{k,i}, \boldsymbol{v}_{k,i}\rangle$. (27)

This procedure converges with linear rate, with the main complexity factor proportional to the $\tau$-th spectral gap (see, e.g.,Theorem 8.2.2 from (Golub & Van Loan, 2013)). More advanced approaches include Oja's and Lanczos iterations (Kuczyński & Woźniakowski, 1992).

**Low-Rank Updates.** The Woodbury matrix identity provides us with the following formula:

$$(\mathbf{H}_k + \alpha_k \mathbf{I})^{-1} = \tfrac{1}{\alpha_k}\big[\mathbf{I} - \mathbf{V}_k\big(\mathbf{I} + \alpha_k \mathrm{Diag}\,(\boldsymbol{a}_k)^{-1}\big)^{-1}\mathbf{V}_k^\top\big].$$

Therefore, we need only $O(\tau^2 n)$ arithmetical operations to perform the step, which is linear with respect to $n$ and can be implemented very efficiently for small $\tau$.

## 8. Experiments

We present illustrative numerical experiments on several machine learning problems. See Section A in the appendix for the details of our experiments and for extra plots.

**Matrix Factorization.** In Fig. 5, we show the convergence of the outer iterations of algorithm (14), solving an ill-conditioned instance of non-convex matrix factorization problems. We study two different strategies of choosing the regularization parameter: constant selection of parameters (left), and using the adaptive search from Section A.2 (right).

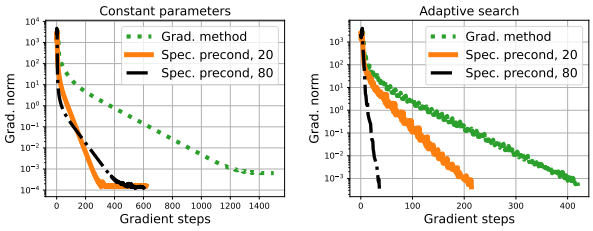


*Figure 5.* Solving an ill-conditioned matrix factorization. Left: best constant parameters, right: adaptive search. Spectral preconditioning helps the methods to converge faster in terms of the gradient steps.

We see that employing the spectral preconditioning helps the method converge faster, in terms of iterations (gradient steps). The use of the adaptive search accelerates the method even further, ensuring an automatic selection of the regularization parameter.

**Logistic Regression.** In the following experiments, we train a convex logistic regression model on several machine learning datasets, using the gradient method with spectral preconditioning. We also compare its performance with quasi-Newton methods: BFGS and the limited memory BFGS (L-BFGS) (Nocedal & Wright, 2006). The results are shown in Fig. 6. We see that employing spectral preconditioning even, for $\tau = 1$ and $\tau = 2$, can significantly accelerate the convergence of the gradient method in terms of both gradient steps and the total number of arithmetic operations. Its overall performance can be comparable to that of the quasi-Newton methods.

## 9. Conclusion

In this work, we propose using inexact top eigenvectors of the Hessian as a preconditioning for gradient methods.

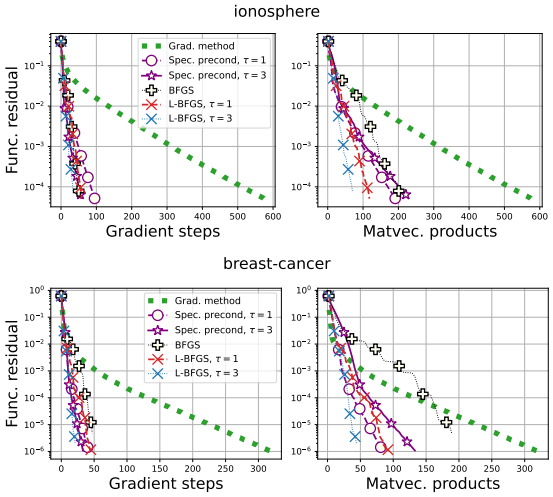


*Figure 6.* Training the logistic regression model. The method with spectral preconditioning works comparably to the BFGS algorithm.

We introduce the notion of *graded non-convexity*, which provides our problem classes with a uniform guarantee on the number of positive eigenvalues. We show that using a preconditioner of order $\tau \geq 1$ provably improves the rate of convergence, cutting the gap between the top $\tau$ eigenvalues.

There are several potential directions for future research. First, it could be interesting to discover whether it is possible to use a Hessian approximation $\mathbf{H}_k$ that has a higher rank than $\tau$, where $\tau$ is the grade of non-convexity, an contains information on directions with negative curvature. In such a case, it seems necessary to use the cubic regularization of trust-region approach to properly bound the length of the method step. At the same time, it poses an interesting open question: can negative curvature be helpful for the method? It is also important to study a behavior of spectral preconditioning in a neighborhood of saddle points.

Another direction is the local analysis of the method. Indeed, besides using $Q = \mathbb{R}^n$ in (7), several other options exist. For example, the initial sublevel set $Q = \{\boldsymbol{x} : f(\boldsymbol{x}) \leq f(\boldsymbol{x}_0)\}$, since, due to Lemma 4.1, all iterations belong to it. Alternatively, explicit regularization can be used with $Q = \{\boldsymbol{x} : \|\boldsymbol{x}\| \leq R\}$. Also, as in the classic analysis of Newton's method, we can assume that the starting point is in a neighborhood $Q$ of a local non-degenerate solution. In this case, the local perspective on graded non-convexity can enhance analysis of quasi-Newton methods.

Finally, it can be interesting to combine our results with recently proposed methods that leverage negative curvature (Carmon et al., 2018). In contrast to the pure first-order methods that typically depend on the leading eigenvalue $\lambda_1$ of the Hessian, in our work we allow for the elimination of this dependence.

## Acknowledgements

This work was supported by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract number 22.00133.

## Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

## References

Bach, F. Self-concordant analysis for logistic regression. 2010.

Carmon, Y., Duchi, J. C., Hinder, O., and Sidford, A. Accelerated methods for nonconvex optimization. *SIAM Journal on Optimization*, 28(2):1751–1772, 2018.

Cartis, C., Gould, N. I., and Toint, P. L. Adaptive cubic regularisation methods for unconstrained optimization. Part I: motivation, convergence and numerical results. *Mathematical Programming*, 127(2):245–295, 2011a.

Cartis, C., Gould, N. I., and Toint, P. L. Adaptive cubic regularisation methods for unconstrained optimization. Part II: worst-case function-and derivative-evaluation complexity. *Mathematical programming*, 130(2):295–319, 2011b.

Cartis, C., Gould, N. I., and Toint, P. L. An example of slow convergence for Newton's method on a function with globally Lipschitz continuous Hessian. Technical report, Technical report, ERGO 13-008, School of Mathematics, Edinburgh University, 2013.

Conn, A. R., Gould, N. I., and Toint, P. L. *Trust region methods*. SIAM, 2000.

Cunha, L., Gidel, G., Pedregosa, F., Scieur, D., and Paquette, C. Only tails matter: Average-case universality and robustness in the convex regime. In *International Conference on Machine Learning*, pp. 4474–4491. PMLR, 2022.

Doikov, N. Minimizing quasi-self-concordant functions by gradient regularization of Newton method. *arXiv preprint arXiv:2308.14742*, 2023.

Doikov, N. and Grapiglia, G. N. First and zeroth-order implementations of the regularized Newton method with lazy approximated Hessians. *arXiv preprint arXiv:2309.02412*, 2023.

Doikov, N. and Nesterov, Y. Gradient regularization of Newton method with Bregman distances. *Mathematical Programming*, pp. 1–25, 2023.

Doikov, N. and Rodomanov, A. Polynomial preconditioning for gradient methods. In *40th International Conference on Machine Learning (ICML 2023)*, number 40, 2023.

Doikov, N., Mishchenko, K., and Nesterov, Y. Super-universal regularized Newton method. *arXiv preprint arXiv:2208.05888*, 2022.

Doikov, N., Chayti, E. M., and Jaggi, M. Second-order optimization with lazy Hessians. In *International Conference on Machine Learning*. PMLR, 2023.

Even, M., Pesme, S., Gunasekar, S., and Flammarion, N. (s) gd over diagonal linear networks: Implicit regularisation, large stepsizes and edge of stability. *arXiv preprint arXiv:2302.08982*, 2023.

Golub, G. H. and Van Loan, C. F. *Matrix computations*. JHU press, 2013.

Goujaud, B., Scieur, D., Dieuleveut, A., Taylor, A. B., and Pedregosa, F. Super-acceleration with cyclical step-sizes. In *International Conference on Artificial Intelligence and Statistics*, pp. 3028–3065. PMLR, 2022.

Grapiglia, G. N. and Nesterov, Y. Regularized Newton methods for minimizing functions with Hölder continuous Hessians. *SIAM Journal on Optimization*, 27(1): 478–506, 2017.

Gromov, M. Sign and geometric meaning of curvature. *Rendiconti del Seminario Matematico e Fisico di Milano*, 61:9–123, 1991.

Hanzely, F., Doikov, N., Nesterov, Y., and Richtarik, P. Stochastic subspace cubic Newton method. In *International Conference on Machine Learning*, pp. 4027–4038. PMLR, 2020.

Hanzely, S. Sketch-and-project meets Newton method: Global O(1/k^2) convergence with low-rank updates. *arXiv preprint arXiv:2305.13082*, 2023.

Hanzely, S., Kamzolov, D., Pasechnyuk, D., Gasnikov, A., Richtárik, P., and Takác, M. A damped Newton method achieves global $\mathcal{O}\left(\frac{1}{k^2}\right)$ and local quadratic convergence rate. *Advances in Neural Information Processing Systems*, 35:25320–25334, 2022.

Hestenes, M. R. and Stiefel, E. Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, 49(6):409–436, 1952.

Hörmander, L. *Notions of convexity*. Springer Science & Business Media, 2007.

Islamov, R., Qian, X., and Richtárik, P. Distributed second order methods with fast rates and compressed communication. In *International conference on machine learning*, pp. 4617–4628. PMLR, 2021.

Islamov, R., Qian, X., Hanzely, S., Safaryan, M., and Richtárik, P. Distributed Newton-type methods with communication compression and Bernoulli aggregation. *arXiv preprint arXiv:2206.03588*, 2022.

Jiang, R. and Mokhtari, A. Accelerated quasi-newton proximal extragradient: Faster rate for smooth convex optimization. *arXiv preprint arXiv:2306.02212*, 2023.

Jin, Q. and Mokhtari, A. Non-asymptotic superlinear convergence of standard quasi-Newton methods. *Mathematical Programming*, 200(1):425–473, 2023.

Kamzolov, D., Ziu, K., Agafonov, A., and Takác, M. Cubic regularized quasi-newton methods. *arXiv preprint arXiv:2302.04987*, 2023.

Karimireddy, S. P., Stich, S. U., and Jaggi, M. Global linear convergence of Newton's method without strong-convexity or Lipschitz gradients. *arXiv preprint arXiv:1806.00413*, 2018.

Kovalev, D., Richtarik, P., Gorbunov, E., and Gasanov, E. Stochastic spectral and conjugate descent methods. *Advances in Neural Information Processing Systems*, 31, 2018.

Kuczyński, J. and Woźniakowski, H. Estimating the largest eigenvalue by the power and Lanczos algorithms with a random start. *SIAM journal on matrix analysis and applications*, 13(4):1094–1122, 1992.

Kunstner, F., Sanches Portella, V., Schmidt, M., and Harvey, N. Searching for optimal per-coordinate step-sizes with multidimensional backtracking. *Advances in Neural Information Processing Systems*, 36, 2024.

Ma, C., Xu, X., Tong, T., and Chi, Y. Provably accelerating ill-conditioned low-rank estimation via scaled gradient descent, even with overparameterization. *arXiv preprint arXiv:2310.06159*, 2023.

Ma, S. and Belkin, M. Diving into the shallows: a computational perspective on large-scale shallow learning. *Advances in neural information processing systems*, 30, 2017.

Mardal, K.-A. and Winther, R. Preconditioning discretizations of systems of partial differential equations. *Numerical Linear Algebra with Applications*, 18(1):1–40, 2011.

Mishchenko, K. Regularized Newton method with global $\mathcal{O}(1/k^2)$ convergence. *SIAM Journal on Optimization*, 33(3):1440–1462, 2023.

Nemirovski, A. and Yudin, D. Problem complexity and method efficiency in optimization. 1983.

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

Nesterov, Y. Implementable tensor methods in unconstrained convex optimization. *Mathematical Programming*, pp. 1–27, 2019.

Nesterov, Y. Quartic regularity. *arXiv preprint arXiv:2201.04852*, 2022.

Nesterov, Y. and Nemirovski, A. *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.

Nesterov, Y. and Polyak, B. T. Cubic regularization of Newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

Nocedal, J. and Wright, S. *Numerical optimization*. Springer Science & Business Media, 2006.

Nutini, J., Laradji, I., and Schmidt, M. Let's make block coordinate descent converge faster: faster greedy rules, message-passing, active-set complexity, and superlinear convergence. *Journal of Machine Learning Research*, 23 (131):1–74, 2022.

Pasechnyuk, D. A., Gasnikov, A., and Takáč, M. Effects of momentum scaling for sgd. *arXiv preprint arXiv:2210.11869*, 2022.

Pasechnyuk, D. A., Gasnikov, A., and Takáč, M. Convergence analysis of stochastic gradient descent with adaptive preconditioning for non-convex and convex functions. *arXiv preprint arXiv:2308.14192*, 2023.

Pesme, S. and Flammarion, N. Saddle-to-saddle dynamics in diagonal linear networks. *arXiv preprint arXiv:2304.00488*, 2023.

Pesme, S., Pillaud-Vivien, L., and Flammarion, N. Implicit bias of sgd for diagonal linear networks: a provable benefit of stochasticity. *Advances in Neural Information Processing Systems*, 34:29218–29230, 2021.

Polyak, B. T. Newton's method and its use in optimization. *European Journal of Operational Research*, 181(3):1086–1096, 2007.

Qian, X., Islamov, R., Safaryan, M., and Richtárik, P. Basis matters: better communication-efficient second order methods for federated learning. *arXiv preprint arXiv:2111.01847*, 2021.

Rodomanov, A. and Kropotov, D. A randomized coordinate descent method with volume sampling. *SIAM Journal on Optimization*, 30(3):1878–1904, 2020.

Rodomanov, A. and Nesterov, Y. New results on superlinear convergence of classical quasi-Newton methods. *Journal of optimization theory and applications*, 188:744–769, 2021a.

Rodomanov, A. and Nesterov, Y. Rates of superlinear convergence for classical quasi-Newton methods. *Mathematical Programming*, pp. 1–32, 2021b.

Safaryan, M., Islamov, R., Qian, X., and Richtárik, P. Fednl: Making Newton-type methods applicable to federated learning. *arXiv preprint arXiv:2106.02969*, 2021.

Scieur, D. and Pedregosa, F. Universal average-case optimality of Polyak momentum. In *International conference on machine learning*, pp. 8565–8572. PMLR, 2020.

Spielman, D. A. and Teng, S.-H. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pp. 81–90, 2004.

Sun, T. and Tran-Dinh, Q. Generalized self-concordant functions: a recipe for Newton-type methods. *Mathematical Programming*, 178(1-2):145–213, 2019.

Vaidya, P. M. Solving linear equations with symmetric diagonally dominant matrices by constructing good preconditioners. *A talk based on this manuscript*, 2(3.4):2–4, 1991.

Vial, J.-P. Strong and weak convexity of sets and functions. *Mathematics of Operations Research*, 8(2):231–259, 1983.

Woodworth, B., Gunasekar, S., Lee, J. D., Moroshko, E., Savarese, P., Golan, I., Soudry, D., and Srebro, N. Kernel and rich regimes in overparametrized models. In Abernethy, J. and Agarwal, S. (eds.), *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 3635–3673. PMLR, 09–12 Jul 2020. URL https://proceedings.mlr.press/v125/woodworth20a.html.

Zhang, G., Fattahi, S., and Zhang, R. Y. Preconditioned gradient descent for overparameterized nonconvex Burer–Monteiro factorization with global optimality certification. *Journal of Machine Learning Research*, 24(163): 1–55, 2023.

Zhang, J., Fattahi, S., and Zhang, R. Y. Preconditioned gradient descent for over-parameterized nonconvex matrix factorization. *Advances in Neural Information Processing Systems*, 34:5985–5996, 2021.

# Appendix

## A. Extra Experiments and Details

In this section, let us provide the details of our experiments and additional numerical results, comparing the performance of the spectral preconditioning.

### A.1. Matrix Factorization

We consider the following non-convex optimization problem, which has applications in dimensionality reduction and clustering:

$$\min_{\substack{\mathbf{X}\in\mathbb{R}^{n\times r}\\ \mathbf{Y}\in\mathbb{R}^{r\times m}}} \Big[ f(\mathbf{X},\mathbf{Y}) \;:=\; \tfrac{1}{2}\|\mathbf{X}\mathbf{Y}-\mathbf{C}\|^2 \Big], \tag{28}$$

where $\mathbf{C}\in\mathbb{R}^{n\times m}$ is a give target matrix and the norm is the standard Frobenius: $\|\mathbf{A}\| := \mathrm{Trace}\,(\mathbf{A}^\top\mathbf{A})^{1/2}$ for $\mathbf{A}\in\mathbb{R}^{n,m}$.

In the following experiment, we fix $n=m=40$ and generate matrix $\mathbf{C}$ by the following procedure: we choose $r^\star := 10$ (the target rank) and set

$$\mathbf{C} \;:=\; \sum_{i=1}^{r^\star} \boldsymbol{u}_i\boldsymbol{v}_i^\top,$$

where $\{\boldsymbol{u}_i\}_{i=1}^{r^\star}$ and $\{\boldsymbol{v}_i\}_{i=1}^{r^\star}$ are generated randomly and uniformly from the unit sphere. Therefore, matrix $\mathbf{C}$ has $r^\star$ singular values that are close to 1, and all others are close to 0. In all our methods we use the initial point $(\mathbf{X}_0,\mathbf{Y}_0)$ generated randomly with entries from the standard normal distribution. A typical distribution of the spectrum of the Hessian of $f$ at the initial point is shown in Fig. 7. We observe that there exist a group of eigenvalues which are significantly larger than the others. Therefore, in such cases, we expect to see an accelerated of the methods by using our spectral preconditioning technique.

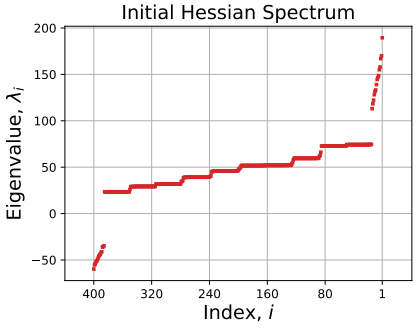


*Figure 7.* Spectrum of the Hessian at the initial point, $r=5$ and $r^\star=10$.

We compare the classic gradient descent with the spectral preconditioning. In the gradient descent, we use the standard Armijo-type line search at each iteration to choose the step size. For the spectral preconditioning, we fix the regularization parameter, according to our theory, at iteration $k\geq 0$:

$$\alpha_k \;:=\; \sqrt{L\|\nabla f(\mathbf{X}_k,\mathbf{Y}_k)\|}+\beta_k,$$

where we fix $L:=1$ and $\beta_k$ is fitted using a simple adaptive search, that ensures sufficient progress of every step (see Section A.2). Namely, we start with an initial value of $\beta_0 := 0.05$. In each iteration, we increase the previous estimate by two until the sufficient condition is satisfied. After performing the method step, we decrease the found value by two. Thus, we allow parameters $\beta_k$ to both increase and decrease with each iteration.

We use one iteration of the hot start power method to update our estimate of $\nabla^2_\tau f$ (see Section 7).

- $r^* = 1$. In this experiment, we compare the performance of spectral preconditioning recovering the matrix $\mathbf{C}$ of rank 1, using $r = 5$ components. The results are shown in Fig. 8.

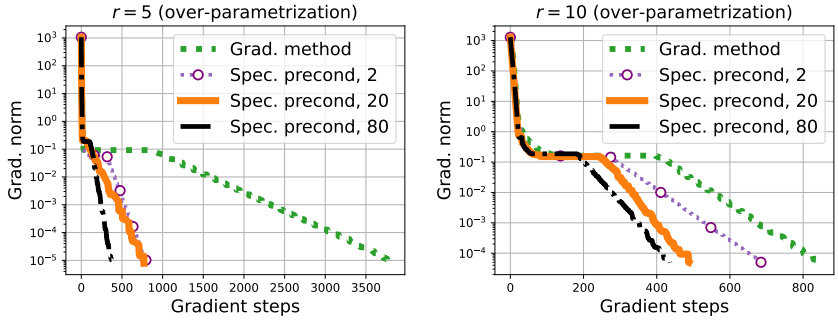


*Figure 8.* Convergence of the methods on the matrix factorization problem, recovering the matrix of rank $r^* = 1$.

- $r^* = 10$. In the following experiment, we test three different settings for the rank parameter: $r < r^\star$ (under-parametrization), $r = r^\star$ (true value), and $r > r^\star$ (over-parametrization). The results are shown in Fig. 9.

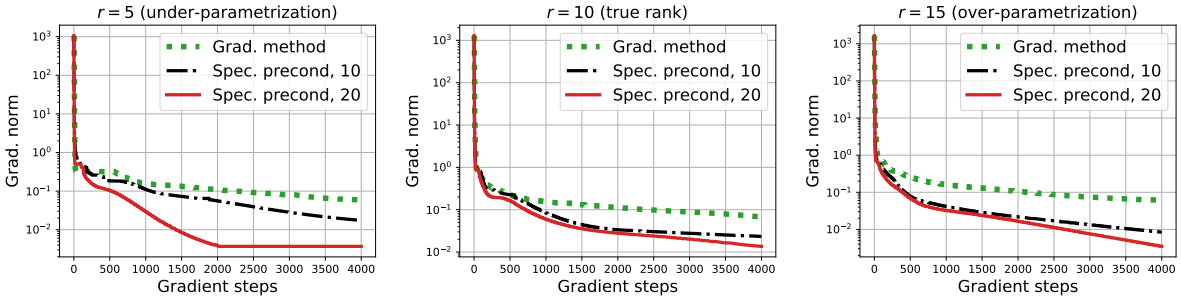


*Figure 9.* Convergence of the methods on the matrix factorization problem, for different values of $r$ (rank estimate), recovering $r^* = 10$.

We see that it becomes easier to solve the problem, when parameter $r$ is increasing. Note that the dimension of problem (28) also growth with $r$. In all cases, using the spectral preconditioning with a small value of $\tau$ accelerates convergence of the method, as predicted by our theory.

### A.2. Selection of the Regularization Parameter

In this section, let us compare two different strategies for choosing the regularization parameter in the spectral preconditioning: a constant choice and adaptive search.

For solving optimization problem $\min_{\boldsymbol{x} \in \mathbb{R}^n} f(\boldsymbol{x})$, we consider the following rule (as in Theorem 4.2), for each iteration $k \geq 0$:

$$\alpha_k \;=\; \sqrt{L \|\nabla f(\boldsymbol{x}_k)\|} + \beta_k.$$

Then, one step of the method is

$$\boldsymbol{x}_{k+1} \;=\; \boldsymbol{x}_k - \left(\mathbf{H}_k + \alpha_k \mathbf{I}\right)^{-1} \nabla f(\boldsymbol{x}_k), \qquad \text{where} \qquad \mathbf{H}_k \approx \nabla^2_\tau f(\boldsymbol{x}_k).$$

Our theory shows that choosing $L$ and $\beta_k$ sufficiently big, we guarantee a significant progress for each iteration (Lemma 4.1), that leads to good global convergence rates. The parameter $L$ is the Lipschitz constant of the Hessian, while $\beta_k$ depends on the $\tau$th spectral gap and the accuracy of approximating the spectral Hessian. Hence, estimating $L$ seems to be easier

**Algorithm 1** Adaptive Gradient Method with Spectral Preconditioning

1: **Input:** $\boldsymbol{x}_0 \in \mathbb{R}^n$. Fix some $L \geq 0$ and $\beta_0 > 0$.
2: **for** $k = 0, 1, \ldots$ **do**
3: Maintain a low-rank approximation $\mathbf{H}_k \approx \nabla^2_\tau f(\boldsymbol{x}_k)$
4: **repeat**
5: Update $\beta_k = 2 \cdot \beta_k$
6: Set $\alpha_k = \sqrt{L\|\nabla f(\boldsymbol{x}_k)\|} + \beta_k$
7: Perform the gradient step: $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \big(\mathbf{H}_k + \alpha_k \mathbf{I}\big)^{-1} \nabla f(\boldsymbol{x}_k)$
8: **until** $f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k+1}) \;\geq\; \frac{1}{8\alpha_k}\|\nabla f(\boldsymbol{x}_{k+1})\|^2$.
9: Set $\beta_{k+1} = \frac{1}{2} \cdot \beta_k$
10: **end for**

than $\beta_k$ (indeed, for some problems, there exist available bounds for the Lipschitz constants, as for example in the logistic regression or in the soft maximum problem). At the same time, estimating parameter $\beta_k$ can be more difficult. Therefore, we use the following adaptive procedure, which fixes the value of $L$ and chooses $\beta_k$ adaptively in each iteration.

This adaptive search resembles a standard Armijo-type procedure, which is popular for the first-order and second-order methods (Nocedal & Wright, 2006; Nesterov, 2018). Importantly, it does not provide the method with any significant overhead, but makes us free from choosing the true value of parameter $\beta_k$.

Note that according to Lemma 4.1, the stopping condition in the inner loop will be satisfied at least when $\beta_k \geq \sigma_\tau + \delta$. Hence, the method is well-defined.

Let us denote by $\beta_k^{\text{init}}$ the first value of parameter $\beta_k$ after its initialization (in line 1 for $\beta_0$ and line 9 for $\beta_{k+1}$). Thus, the number of times the inner loop in lines 4-8 is performed at iteration $k \geq 0$ is $1 + \log_2(\beta_{k+1}^{\text{init}} / \beta_k^{\text{init}})$ and the total number $T_k$ of the gradient steps during the first $k$ iterations of the algorithm is estimated as

$$\begin{aligned} T_k \;&=\; \sum_{i=0}^{k-1} (1 + \log_2 \tfrac{\beta_{i+1}^{\text{init}}}{\beta_i^{\text{init}}}) \;=\; k + \log_2 \prod_{i=0}^{k-1} \tfrac{\beta_{i+1}^{\text{init}}}{\beta_i^{\text{init}}} \;=\; k + \log_2 \beta_k^{\text{init}} - \log \beta_0^{\text{init}} \\ &\leq\; k + 1 + \log_2 \tfrac{\sigma_\tau + \delta}{\beta_0^{\text{init}}}, \end{aligned}$$

where we used in the last bound that $\beta_k \leq \max\{2(\sigma_\tau + \delta), \beta_0^{\text{init}}\}$, $\forall k \geq 0$, due to Lemma 4.1. Therefore, up to an additional logarithmic term, the adaptive search method performs the same number of the gradient steps as the method with a fixed (known) value of the regularization constant. In average, it requires only one extra gradient computation (line 8) per iteration.

In the following graphs, let us compare two strategies for choosing the regularization parameter:

- Constant choice (fitted separately for each method and each problem);
- Adaptive search, provided by Algorithm 1.

We test the matrix factorization problem as in the previous experiments:

$$\min_{\substack{\mathbf{X} \in \mathbb{R}^{n \times r} \\ \mathbf{Y} \in \mathbb{R}^{r \times m}}} \Big[ f(\mathbf{X}, \mathbf{Y}) \;\;:=\;\; \tfrac{1}{2}\|\mathbf{X}\mathbf{Y} - \mathbf{C}\|^2 \Big], \tag{29}$$

where $\mathbf{C} \in \mathbb{R}^{n \times m}$ is a give target matrix, generated randomly, of a fixed rank $r^\star$. We set $n = m = 40$ and use the initial points $(\mathbf{X}_0, \mathbf{Y}_0)$ generated randomly with entries from the standard normal distribution.

**Ill-conditioned problem.** We consider an ill-conditioned problem of rank $r^\star = 1$. The corresponding singular value of matrix $\mathbf{C}$ is $400$.

In the following graphs, we see convergence of the methods for the constant selection of parameters (left), and using the adaptive search (right). We observe that not all values of the gradient norm are monotonically decreasing. This does

not contradict our theory, since the guarantees given by Theorem 4.2 and Theorem 5.1 are given in terms of the *minimal gradient norm*. In Figure 5, we show the convergence in terms of the *current gradient norm* $\|\nabla f(\boldsymbol{x}_k)\|$, and in Figure 10, we demonstrate the convergence of the same methods for the *minimal gradient norm* $\min\limits_{0\leq i\leq k} \|\nabla f(\boldsymbol{x}_i)\|$, for each iteration $k \geq 0$.

We see that using the adaptive search might increase oscillations of the gradient norm. However, it also helps to reach the desired value of the minimal gradient norm faster. We find that using adaptive search is always helpful, as the method becomes independent of the parameters $\sigma_\tau$ and $\delta$.

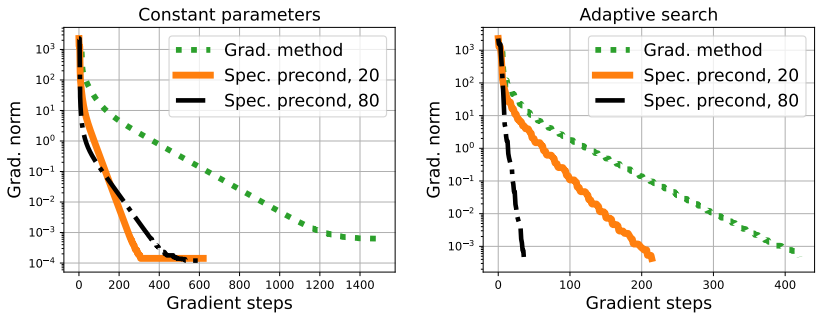


*Figure 10.* The graphs from Figure 5 for the minimal gradient norm $\min\limits_{0\leq i\leq k} \|\nabla f(\boldsymbol{x}_i)\|$.

### A.3. Convex Models

In the following numerical experiments we train a logistic regression model on several machine learning datasets[5]. Thus, our optimization problem has the following form:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \Big[ f(\boldsymbol{x}) \quad := \quad \frac{1}{m} \sum_{i=1}^{m} \ln(1 + e^{\langle \boldsymbol{a}_i, \boldsymbol{x} \rangle}) \Big], \tag{30}$$

where vectors $\{\boldsymbol{a}_i\}_{i=1}^m$ are determined by the dataset.

The results are shown in Fig. 11. We see that the spectral preconditioning works significantly better than the basic gradient descent and comparable with the powerful BFGS and the limited memory BFGS (L-BFGS) methods (Nocedal & Wright, 2006). The spectral preconditioning usually outperforms BFGS algrorithm, for a low accuracy level, and works slightly worse than L-BFGS, using the same number of $\tau$ directions (the memory size).

The classical quasi-Newton methods could achieve better practical performance in some cases because of their superlinear local convergence (Rodomanov & Nesterov, 2021a). In contrast, the spectral preconditioning is designed to provide strong global convergence guarantees, including those for non-convex problems. Therefore, developing a low-rank spectral preconditioning technique equipped with local superlinear rates could be an interesting direction for future research.

[5] https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

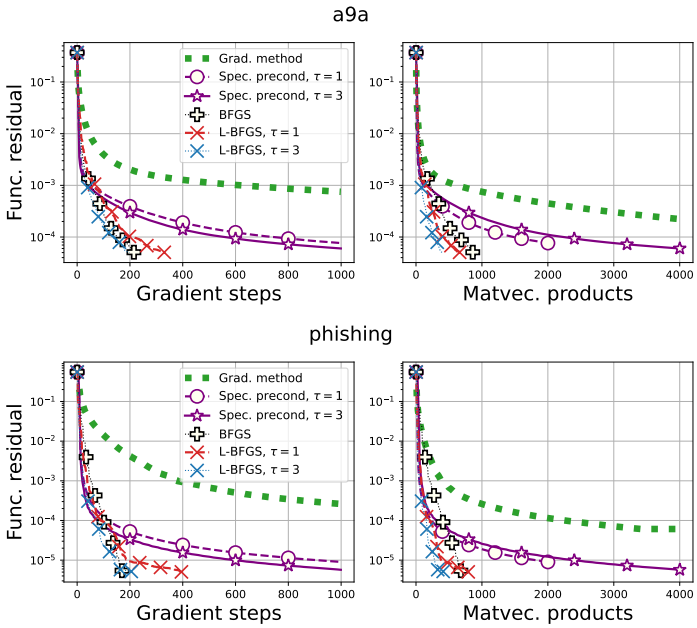


*Figure 11.* Training logistic regression using the spectral preconditioning with $\tau = 1$ and $\tau = 3$. We see that the spectral preconditioning works significantly better than the basic gradient descent and comparable with the powerful BFGS algorithm.

## B. Proofs for Section 2

### B.1. Proof of Proposition 2.2

*Proof.* For any two symmetric matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$, and $1 \le i, j \le n$ such that $i + j \ge n + 1$, we have the following Weyl's inequality:

$$\lambda_{i+j-n}(\mathbf{A}+\mathbf{B}) \quad \ge \quad \lambda_i(\mathbf{A}) + \lambda_j(\mathbf{B}), \tag{31}$$

which immediately proves the statement about the sum. To prove the statement for soft maximum, let us denote

$$f(\boldsymbol{x}) \quad := \quad \operatorname{smax}(f_1(\boldsymbol{x}), f_2(\boldsymbol{x})) \quad = \quad \ln(e^{f_1(\boldsymbol{x})} + e^{f_2(\boldsymbol{x})}),$$

and compute the derivatives. We obtain, for any $\boldsymbol{h} \in \mathbb{R}^n$:

$$\langle \nabla f(\boldsymbol{x}), \boldsymbol{h}\rangle \quad = \quad \sigma(f_1(\boldsymbol{x}) - f_2(\boldsymbol{x})) \cdot \langle \nabla f_1(\boldsymbol{x}), \boldsymbol{h}\rangle + \sigma(f_2(\boldsymbol{x}) - f_1(\boldsymbol{x})) \cdot \langle \nabla f_2(\boldsymbol{x}), \boldsymbol{h}\rangle,$$

where $\sigma(t) \stackrel{\text{def}}{=} \frac{1}{1+e^{-t}}$. Note that $\sigma'(t) = \sigma(t)\cdot\sigma(-t)$. Denoting $\sigma_1 \stackrel{\text{def}}{=} \sigma(f_1(\boldsymbol{x}) - f_2(\boldsymbol{x}))$, $\sigma_2 \stackrel{\text{def}}{=} \sigma(f_1(\boldsymbol{x}) - f_2(\boldsymbol{x}))$, we get

$$\begin{aligned}\langle \nabla^2 f(\boldsymbol{x})\boldsymbol{h}, \boldsymbol{h}\rangle \quad &= \quad \sigma_1\sigma_2\Big[\langle \nabla f_1(\boldsymbol{x}), \boldsymbol{h}\rangle - \langle \nabla^2 f(\boldsymbol{x}), \boldsymbol{h}\rangle\Big]^2 + \sigma_1\langle \nabla^2 f_1(\boldsymbol{x})\boldsymbol{h}, \boldsymbol{h}\rangle + \sigma_2\langle \nabla^2 f_2(\boldsymbol{x})\boldsymbol{h}, \boldsymbol{h}\rangle \\ &\ge \quad \sigma_1\langle \nabla^2 f_1(\boldsymbol{x})\boldsymbol{h}, \boldsymbol{h}\rangle + \sigma_2\langle \nabla^2 f_2(\boldsymbol{x})\boldsymbol{h}, \boldsymbol{h}\rangle.\end{aligned}$$

Using (31) completes the proof. □

### B.2. Proof of Proposition 2.3

*Proof.* The Hessian of $g$ is given by

$$\nabla^2 g(\boldsymbol{x}) \quad = \quad \mathbf{A}^\top \nabla^2 f(L(\boldsymbol{x}))\mathbf{A} \quad \in \quad \mathbb{R}^{m\times m}.$$

The linear map $\mathbf{A} : \mathbb{R}^m \to \mathbb{R}^n$ induces the isomorphism: $\mathbb{R}^m/\operatorname{Ker}(\mathbf{A}) \stackrel{\varphi}{\to} \operatorname{Im}(\mathbf{A})$ and thus $m = \dim\operatorname{Ker}(\mathbf{A}) + \dim\operatorname{Im}(\mathbf{A})$ (the classical Rank–nullity theorem).

By our assumption, $\nabla^2_\tau f(L(\boldsymbol{x})) \succeq \mathbf{0}$. Let us denote by $V_+$ the subspace spanned by top $\tau$ eigenvectors of $\nabla^2 f(L(\boldsymbol{x}))$, and $V_-$ is spanned by the rest. Hence, $\mathbb{R}^n = V_+ \oplus V_-$. Then

$$\begin{aligned} m \quad &= \quad \dim\operatorname{Ker}(\mathbf{A}) + \dim\operatorname{Im}(\mathbf{A}) \cap V_+ + \dim\operatorname{Im}(\mathbf{A}) \cap V_- \\ &\le \quad \dim\operatorname{Ker}(\mathbf{A}) + \dim\operatorname{Im}(\mathbf{A}) \cap V_+ + n - \tau.\end{aligned}$$

Rearranging the terms, we get

$$\dim\operatorname{Ker}(\mathbf{A}) + \dim\operatorname{Im}(\mathbf{A}) \cap V_+ \quad \ge \quad m - n + \tau.$$

Therefore, we conclude that the linear subspace $U = \operatorname{Ker}(\mathbf{A}) \oplus \varphi^{-1}(\operatorname{Im}(\mathbf{A}) \cap V_+)$ has dimension $\dim U \ge m - n + \tau$, and for any $\boldsymbol{h} \in U$:

$$\langle \nabla^2 g(\boldsymbol{x})\boldsymbol{h}, \boldsymbol{h}\rangle \quad \ge \quad 0,$$

which proves the required bound. □

### B.3. Proof of Proposition 2.4

*Proof.* For a small $\varepsilon > 0$, we consider the regularized objective

$$g(\boldsymbol{x}) \quad := \quad f(\boldsymbol{x}) + \tfrac{\varepsilon}{2}\|\boldsymbol{x}\|^2.$$

Assume that $g$ achieves its maximum $\boldsymbol{x}^\star_g = \arg\max_{\boldsymbol{x}\in K} g(\boldsymbol{x})$ in the interior, $\boldsymbol{x}^\star_g \in \operatorname{int} K$. Then, the second-order stationary condition implies that

$$\nabla^2 g(\boldsymbol{x}^\star_g) \quad \preceq \quad \mathbf{0},$$

which is impossible due to $\lambda_1(\nabla^2 g(\boldsymbol{x}^\star_g)) = \lambda_1(\nabla^2 f(\boldsymbol{x}^\star_g)) + \varepsilon > 0$. Hence, $\boldsymbol{x}^\star_g \in \partial K$ and we have that

$$\max_{\boldsymbol{x}\in K} f(\boldsymbol{x}) \quad \le \quad \max_{\boldsymbol{x}\in K} g(\boldsymbol{x}) \quad = \quad \max_{\boldsymbol{x}\in \partial K} g(\boldsymbol{x}) \quad \le \quad \max_{\boldsymbol{x}\in \partial K} f(\boldsymbol{x}) + \tfrac{\varepsilon}{2}R^2,$$

where $R$ is the radius of a ball containing $K$. Tending $\varepsilon$ to 0 completes the proof. □

### B.4. Proof of Proposition 2.5

*Proof.* Indeed, by the Taylor theorem, we have that for any $\boldsymbol{h} \in V_{\boldsymbol{x}}$ and $\lambda > 0$ there exists $\xi \in [0, 1]$ such that

$$\begin{aligned} 0 &\leq f(\boldsymbol{x} + \lambda \boldsymbol{h}) - f(\boldsymbol{x}) - \lambda \langle \nabla f(\boldsymbol{x}), \boldsymbol{h} \rangle \\ &= \tfrac{\lambda^2}{2} \langle \nabla^2 f(\boldsymbol{x} + \lambda \xi \boldsymbol{h}) \boldsymbol{h}, \boldsymbol{h} \rangle. \end{aligned}$$

Dividing both sides by $\lambda^2$ and taking the limit $\lambda \to 0$, we obtain that

$$\langle \nabla^2 f(\boldsymbol{x}) \boldsymbol{h}, \boldsymbol{h} \rangle \geq 0, \qquad \forall \boldsymbol{h} \in V_{\boldsymbol{x}}.$$

Therefore, since $\dim(V_{\boldsymbol{x}}) \geq \tau$, we have that $\lambda_\tau(\boldsymbol{x}) \geq 0$. □

## C. Composite Optimization and Proofs for Section 3

### C.1. Composite Formulation

The main results of our work can be generalized to a more broad family of Composite Optimization Problems of the following form:

$$\min_{\boldsymbol{x} \in Q} \Big[ F(\boldsymbol{x}) \;:=\; f(\boldsymbol{x}) + \psi(\boldsymbol{x}) \Big], \tag{32}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is our smooth objective (which can be non-convex), and $\psi : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a *simple* closed convex function (e.g., indicator of a given closed convex set, or a regularizer). We set $Q := \operatorname{dom} \psi \subseteq \mathbb{R}^n$. Thus, the main properties of $f$ (the grade of non-convexity and the level of smoothness can be defined with respect to this convex set $Q$).

In case of presence of the composite part $\psi(\cdot)$ in our problem, iterations of our method should be modified. For a given point $\boldsymbol{x} \in Q$, gradient vector $\boldsymbol{g} \in \mathbb{R}^n$ and matrix $\mathbf{H} = \mathbf{H}^\top \succeq 0$, we define the composite gradient step, as follows:

$$\text{CompositeStep}_{\mathbf{H},\alpha}(\boldsymbol{x}, \boldsymbol{g}) \;:=\; \operatorname*{argmin}_{\boldsymbol{y} \in Q} \Big\{ \langle \boldsymbol{g}, \boldsymbol{y} \rangle + \psi(\boldsymbol{y}) + \tfrac{1}{2} \langle (\mathbf{H} + \alpha \mathbf{I})(\boldsymbol{y} - \boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle \Big\}, \tag{33}$$

where $\alpha \geq 0$ is the regularization parameter. In general, due to the presence of regularization term in (33), for $\alpha > 0$, the composite subproblem is *strongly convex*, and we can employ the fast linear convergence of gradient methods as applied to (33), for computing this step inexactly.

With these modifications, we are ready to present a composite version of our algorithm for solving (32):

**Composite Gradient Method with Spectral Preconditioning** (34)

**Choose** $\boldsymbol{x}_0 \in Q$ and $0 \leq \tau \leq n$.

**For** $k \geq 0$ **iterate:**

1. Estimate $\mathbf{H}_k \approx \nabla^2_\tau f(\boldsymbol{x}_k) \in \mathbb{R}^{n \times n}$
2. Perform the composite gradient step, for some $\alpha_k \geq 0$:

$$\boldsymbol{x}_{k+1} = \text{CompositeStep}_{\mathbf{H}_k, \alpha_k}(\boldsymbol{x}_k, \nabla f(\boldsymbol{x}_k))$$

Note that when $\psi \equiv 0$, composite step (33) coincides with our basic preconditioned gradient step defined earlier:

$$\text{CompositeStep}_{\mathbf{H},\alpha}(\boldsymbol{x}, \boldsymbol{g}) = \text{GradStep}_{\mathbf{H},\alpha}(\boldsymbol{x}, \boldsymbol{g}) = \boldsymbol{x} - \big(\mathbf{H} + \alpha \mathbf{I}\big)^{-1} \mathbf{g},$$

and method (34) is identical to (14).

### C.2. Convergence Analysis

In this section, we provide the proofs of our main convergence results from Section 3. We study the more general composite formulation (34) of our method, which covers the basic case when $\psi \equiv 0$.

Let us consider one step of our method: $\boldsymbol{x}^+ = \text{CompositeStep}_{\mathbf{H},\alpha}(\boldsymbol{x}, \nabla f(\boldsymbol{x}))$, for some $\alpha > 0$, and establish its key properties. The new point $\boldsymbol{x}^+$ satisfies the following optimality condition (see, e.g., Theorem 3.1.23 in (Nesterov, 2018)):

$$\langle \nabla f(\boldsymbol{x}) + (\mathbf{H} + \alpha \mathbf{I})(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x}^+ \rangle + \psi(\boldsymbol{y}) \quad \geq \quad \psi(\boldsymbol{x}^+), \qquad \forall \boldsymbol{y} \in Q. \tag{35}$$

In other words, the vector $\psi'(\boldsymbol{x}^+) := -\nabla f(\boldsymbol{x}) - (\mathbf{H} + \alpha \mathbf{I})(\boldsymbol{x}^+ - \boldsymbol{x})$ belongs to the subdifferential of $\psi$ at new point:

$$\psi'(\boldsymbol{x}^+) \quad \in \quad \partial \psi(\boldsymbol{x}^+).$$

We denote

$$F'(\boldsymbol{x}^+) \quad := \quad \nabla f(\boldsymbol{x}^+) + \psi'(\boldsymbol{x}^+), \tag{36}$$

that is the main object for which we prove the convergence of our method. Utilizing positive semi-definitiveness of $\mathbf{H} = \mathbf{H}^\top \succeq 0$, we can can bound the length of our displacement $r \stackrel{\text{def}}{=} \|\boldsymbol{x}^+ - \boldsymbol{x}\|$.

**Lemma C.1.** *For any $\boldsymbol{s} \in \partial \psi(\boldsymbol{x})$, we have*

$$r \quad \leq \quad \tfrac{\|\nabla f(\boldsymbol{x}) + \boldsymbol{s}\|}{\alpha}, \tag{37}$$

*and*

$$\langle \mathbf{H}(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x} \rangle \quad \leq \quad r \|\nabla f(\boldsymbol{x}) + \boldsymbol{s}\|. \tag{38}$$

*Proof.* Indeed, using convexity of $\psi$, we obtain, for any $\boldsymbol{s} \in \partial \psi(\boldsymbol{x})$ and our specific $\psi'(\boldsymbol{x}^+) \in \partial \psi(\boldsymbol{x}^+)$:

$$\begin{aligned} 0 \quad &\leq \quad \langle \psi'(\boldsymbol{x}^+) - \boldsymbol{s}, \boldsymbol{x}^+ - \boldsymbol{x} \rangle \\ &= \quad \langle -\nabla f(\boldsymbol{x}) - (\mathbf{H} + \alpha \mathbf{I})(\boldsymbol{x}^+ - \boldsymbol{x}) - \boldsymbol{s}, \boldsymbol{x}^+ - \boldsymbol{x} \rangle \\ &= \quad \langle \nabla f(\boldsymbol{x}) + \boldsymbol{s}, \boldsymbol{x} - \boldsymbol{x}^+ \rangle - \langle (\mathbf{H} + \alpha \mathbf{I})(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x} \rangle. \end{aligned}$$

Rearranging the terms and using Cauchy-Schwartz inequality, we get

$$r \|\nabla f(\boldsymbol{x}) + \boldsymbol{s}\| \quad \geq \quad \langle \nabla f(\boldsymbol{x}) + \boldsymbol{s}, \boldsymbol{x} - \boldsymbol{x}^+ \rangle \quad \geq \quad \langle \mathbf{H}(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x} \rangle + \alpha r^2.$$

Taking into account that $\mathbf{H} \succeq \mathbf{0}$ completes the proof. □

Therefore, by choosing regularization parameter $\alpha$ appropriately, we can control the length of steps for our algorithm. When combined with smoothness properties (17), (18) of the objective, we can establish the global progress in terms of the objective function value. We denote

$$\delta \quad := \quad \|\mathbf{H} - \nabla^2_\tau f(\boldsymbol{x})\|.$$

**Lemma C.2.** *Let $\boldsymbol{s} \in \partial \psi(\boldsymbol{x})$ be some subgradient of $\psi$ at current point $\boldsymbol{x}$ and let $\alpha \geq \sqrt{\frac{L\|\nabla f(\boldsymbol{x}) + \boldsymbol{s}\|}{3}} + \sigma_\tau + \delta$. Then*

$$F(\boldsymbol{x}) - F(\boldsymbol{x}^+) \quad \geq \quad \tfrac{\alpha}{2} r^2. \tag{39}$$

*Proof.* By Lipschitz continuity of the Hessian, we have

$$\begin{aligned} F(\boldsymbol{x}^+) \quad &= \quad f(\boldsymbol{x}^+) + \psi(\boldsymbol{x}^+) \\ &\leq \quad f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x} \rangle + \tfrac{1}{2} \langle \nabla^2 f(\boldsymbol{x})(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x} \rangle + \tfrac{L}{6} r^3 + \psi(\boldsymbol{x}^+) \\ &\overset{(37)}{\leq} \quad f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x} \rangle + \tfrac{1}{2} \langle \nabla^2 f(\boldsymbol{x})(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x} \rangle + \tfrac{L\|\nabla f(\boldsymbol{x}) + \boldsymbol{s}\|}{6\alpha} r^2 + \psi(\boldsymbol{x}^+) \\ &\leq \quad f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x} \rangle + \tfrac{1}{2} \langle \mathbf{H}(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x} \rangle + \tfrac{1}{2} \big(\sigma_\tau + \delta + \tfrac{L\|\nabla f(\boldsymbol{x}) + \boldsymbol{s}\|}{3\alpha}\big) r^2 + \psi(\boldsymbol{x}^+), \end{aligned}$$

where $\sigma_\tau := \sup_{\boldsymbol{x}} \|\nabla^2 f(\boldsymbol{x}) - \nabla^2_\tau f(\boldsymbol{x})\|$ as defined in (18). According to (35), it holds

$$\psi(\boldsymbol{x}^+) \;\leq\; \psi(\boldsymbol{x}) - \langle \nabla f(\boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x}\rangle - \langle \mathbf{H}(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x}\rangle - \alpha r^2.$$

Hence, substituting this inequality into the previous one, we can continue as follows:

$$\begin{aligned} F(\boldsymbol{x}^+) \;&\leq\; F(\boldsymbol{x}) - \tfrac{\alpha}{2} r^2 - \tfrac{1}{2}\big(\alpha - \sigma_\tau - \delta - \tfrac{L\|\nabla f(\boldsymbol{x})+\boldsymbol{s}\|}{3\alpha}\big) r^2 - \tfrac{1}{2}\langle \mathbf{H}(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x}\rangle \\ &\leq\; F(\boldsymbol{x}) - \tfrac{\alpha}{2} r^2 - \tfrac{1}{2}\big(\alpha - \sigma_\tau - \delta - \tfrac{L\|\nabla f(\boldsymbol{x})+\boldsymbol{s}\|}{3\alpha}\big) r^2. \end{aligned}$$

To prove the result, it suffices to check $\alpha \geq \sigma_\tau + \delta + \frac{L\|\nabla f(\boldsymbol{x})+\boldsymbol{s}\|}{3\alpha}$, which is ensured by our choice. Indeed, denoting $B := \sigma_\tau + \delta$ and $C := \frac{L\|\nabla f(\boldsymbol{x})+s\|}{2}$, the inequality which we need to ensure for $\alpha > 0$ is

$$\alpha^2 \;\geq\; B + \tfrac{C}{\alpha} \quad\Leftrightarrow\quad \alpha^2 - \alpha B - C \;\geq\; 0 \quad\Leftrightarrow\quad \alpha \;\geq\; \tfrac{B+\sqrt{B^2+4C}}{2}. \tag{40}$$

It is immediate to check that $\alpha := B + \sqrt{C}$ satisfies (40), hence it holds for any $\alpha \geq B + \sqrt{C}$, which is our choice in the condition of the lemma. □

Now, let us relate the length $r$ of the step with the norm of vector $F'(\boldsymbol{x}^+) := \nabla f(\boldsymbol{x}^+) + \psi'(\boldsymbol{x}^+)$.

**Lemma C.3.** *Let $\boldsymbol{s} \in \partial\psi(\boldsymbol{x})$ be some subgradient of $\psi$ at current point $\boldsymbol{x}$ and let $\alpha \geq \sqrt{\frac{L\|\nabla f(\boldsymbol{x})+\boldsymbol{s}\|}{2}} + \sigma_\tau + \delta$. Then*

$$r \;\geq\; \tfrac{1}{2\alpha}\|F'(\boldsymbol{x}^+)\|. \tag{41}$$

*Proof.* Using the definition of $\psi'(\boldsymbol{x}^+) := -\nabla f(\boldsymbol{x}) - (\mathbf{H} + \alpha\mathbf{I})(\boldsymbol{x}^+ - \boldsymbol{x})$ and Lipschitz continuity of the Hessian, we have

$$\begin{aligned} \|F'(\boldsymbol{x}^+)\| \;&=\; \|\nabla f(\boldsymbol{x}^+) + \psi'(\boldsymbol{x}^+)\| \\ &=\; \|\nabla f(\boldsymbol{x}^+) - \nabla f(\boldsymbol{x}) - (\mathbf{H} + \alpha\mathbf{I})(\boldsymbol{x}^+ - \boldsymbol{x})\| \\ &\leq\; \|\nabla f(\boldsymbol{x}^+) - \nabla f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{x})(\boldsymbol{x}^+ - \boldsymbol{x})\| + \big(\alpha + \sigma_\tau + \delta\big) r \\ &\overset{(17)}{\leq}\; \tfrac{L}{2} r^2 + \big(\alpha + \sigma_\tau + \delta\big) r \\ &\overset{(37)}{\leq}\; \big(\alpha + \sigma_\tau + \delta + \tfrac{L\|\nabla f(\boldsymbol{x})+\boldsymbol{s}\|}{2\alpha}\big) r \;\leq\; 2\alpha r, \end{aligned}$$

where the last inequality holds due to $\alpha \geq \sigma_\tau + \delta + \frac{L\|\nabla f(\boldsymbol{x})+\boldsymbol{s}\|}{2\alpha}$, which is ensure by our choice (see the end of the proof of Lemma C.2). □

Therefore, combining these two Lemmas together, we obtain the following bound for one step of the method.

**Corollary C.4.** *Let $\alpha \geq \sqrt{\frac{L\|\nabla f(\boldsymbol{x})+\boldsymbol{s}\|}{2}} + \sigma_\tau + \delta$ for some $\boldsymbol{s} \in \partial\psi(\boldsymbol{x})$. Then*

$$F(\boldsymbol{x}) - F(\boldsymbol{x}^+) \;\overset{(39),(41)}{\geq}\; \tfrac{1}{8\alpha}\|F'(\boldsymbol{x}^+)\|^2. \tag{42}$$

We are ready to prove the global complexity bound for convergence of our method.

**Theorem C.5.** *Let $f \in \mathcal{F}_\tau$ be non-convex of grade $\tau$, where $0 \leq \tau \leq n$ is fixed. Let $f$ have a Lipschitz Hessian with constant $L$ and bounded parameter $\sigma_\tau$. Consider iterations $\{\boldsymbol{x}_k\}_{k\geq 0}$ of algorithm (34) with*

$$\alpha_k \;=\; \sqrt{\tfrac{L\|F'(\boldsymbol{x}_k)\|}{2}} + \sigma_\tau + \delta,$$

*where $F'(\boldsymbol{x}_k)$ is defined by (36) for $k \geq 1$, and $F'(\boldsymbol{x}_0) := \nabla f(\boldsymbol{x}_0) + \boldsymbol{s}$, for an arbitrary initial subgradient $\boldsymbol{s} \in \partial\psi(\boldsymbol{x}_0)$. Then, for any $\varepsilon > 0$, it is enough to do*

$$K \;=\; \Big\lceil 8(F(\boldsymbol{x}_0) - F^\star) \cdot \Big(\sqrt{\tfrac{L}{2}}\tfrac{1}{\varepsilon^{3/2}} + \tfrac{\sigma_\tau+\delta}{\varepsilon^2}\Big) \;+\; 2\ln\tfrac{\|F'(\boldsymbol{x}_0)\|}{\varepsilon}\Big\rceil. \tag{43}$$

*iterations to have $\min\limits_{1\leq i\leq K} \|F'(\boldsymbol{x}_i)\| \leq \varepsilon$.*

*Proof.* Let us fix some $k \geq 0$ and assume that for any $0 \leq i \leq k$ we have $g_i \overset{\text{def}}{=} \|F'(\boldsymbol{x}_i)\| \geq \varepsilon$.

According to (42), we obtain, for $0 \leq i \leq k-1$:

$$\begin{aligned} F(\boldsymbol{x}_i) - F(\boldsymbol{x}_{i+1}) \quad &\geq \quad \tfrac{1}{8\alpha_k} g_{i+1}^2 \quad = \quad \tfrac{1}{8}\left[\tfrac{g_{i+1}}{g_i}\right]^2 \cdot \left(\sqrt{\tfrac{L}{2}}\tfrac{1}{g_{i+1}^{3/2}} + \tfrac{\sigma_\tau + \delta}{g_{i+1}^2}\right)^{-1} \\ &\geq \quad \tfrac{1}{8}\left[\tfrac{g_{i+1}}{g_i}\right]^2 \cdot \left(\sqrt{\tfrac{L}{2}}\tfrac{1}{\varepsilon^{3/2}} + \tfrac{\sigma_\tau + \delta}{\varepsilon^2}\right)^{-1}. \end{aligned} \tag{44}$$

Denote

$$c \quad := \quad \tfrac{1}{8}\left(\sqrt{\tfrac{L}{2}}\tfrac{1}{\varepsilon^{3/2}} + \tfrac{\sigma_\tau + \delta}{\varepsilon^2}\right)^{-1}.$$

Then, telescoping bound (44) and using the inequality between arithmetic and geometric means, we get

$$\begin{aligned} F(\boldsymbol{x}_0) - F^\star \quad &\geq \quad F(\boldsymbol{x}_0) - F(\boldsymbol{x}_k) \quad \overset{(44)}{\geq} \quad c\sum_{i=0}^{k-1}\left[\tfrac{g_{i+1}}{g_i}\right]^2 \quad \geq \quad ck\left[\prod_{i=0}^{k-1}\tfrac{g_{i+1}}{g_i}\right]^{2/k} \\ &= \quad ck\left[\tfrac{g_k}{g_0}\right]^{2/k} \quad \geq \quad ck\left[\tfrac{\varepsilon}{g_0}\right]^{2/k} \quad = \quad ck\exp\left[-\tfrac{2}{k}\ln\tfrac{g_0}{\varepsilon}\right] \\ &\geq \quad ck\left[1 - \tfrac{2}{k}\ln\tfrac{g_0}{\varepsilon}\right] \quad = \quad ck - 2c\ln\tfrac{g_0}{\varepsilon}. \end{aligned}$$

Hence, we obtain

$$k \quad \leq \quad \tfrac{F(\boldsymbol{x}_0) - F^\star}{c} + 2\ln\tfrac{g_0}{\varepsilon}.$$

Substituting the value of $c$ completes the proof. □

### C.3. Proof of Theorem 4.2

*Proof.* It follows immediately from Theorem C.5 by substituting the non-composite case $\psi \equiv 0$. □

## D. Proofs for Section 5

The results of this section are applied to a basic unconstrained minimization problem (4). Let us consider one iteration $\boldsymbol{x} \mapsto \boldsymbol{x}^+$ of our method, which satisfies the following stationary condition

$$\nabla f(\boldsymbol{x}) + \mathbf{H}(\boldsymbol{x}^+ - \boldsymbol{x}) + \alpha\mathbf{I} \quad = \quad \mathbf{0}, \tag{45}$$

where regularization parameter is chosen as

$$\alpha \quad := \quad \alpha^\star + \eta, \tag{46}$$

with

$$\alpha^\star \quad := \quad \operatorname*{argmax}_{\alpha > 0}\left[-\tfrac{1}{2}\langle(\mathbf{H} + (\alpha+\eta)\mathbf{I})^{-1}\nabla f(\boldsymbol{x}), \nabla f(\boldsymbol{x})\rangle - \tfrac{2\alpha^3}{3L}\right], \tag{47}$$

and $\eta := \sigma_\tau^+ + \delta + \delta_- \geq 0$ is the balancing term to control the errors. Considering the decomposition of the Hessian onto positive and negative components:

$$\nabla^2 f(\boldsymbol{x}) \quad \equiv \quad \nabla_+^2 f(\boldsymbol{x}) - \nabla_-^2 f(\boldsymbol{x}), \qquad \nabla_+^2 f(\boldsymbol{x}),\, \nabla_-^2 f(\boldsymbol{x}) \quad \succeq \quad \mathbf{0}, \tag{48}$$

we have that

$$\begin{aligned} \|\nabla_+^2 f(\boldsymbol{x}) - \nabla_\tau^2 f(\boldsymbol{x})\| \quad &\leq \quad \sigma_\tau^+ \\ \|\mathbf{H} - \nabla_\tau^2 f(\boldsymbol{x})\| \quad &\leq \quad \delta, \\ \|\nabla_-^2 f(\boldsymbol{x})\mathbf{P}\| \quad &\leq \quad \delta_-, \end{aligned} \tag{49}$$

where $\mathbf{P}$ is a given projector onto image of $\mathbf{H}$, that satisfies

$$\mathbf{H}\mathbf{P} \quad = \quad \mathbf{H}. \tag{50}$$

First, let us provide another description of regularization parameter $\alpha^\star$, that is more suitable for our analysis. First-order stationary condition for (47) gives

$$\tfrac{1}{2}\langle(\mathbf{H}+\alpha\mathbf{I})^{-1}\nabla f(\boldsymbol{x}),(\mathbf{H}+\alpha\mathbf{I})^{-1}\nabla f(\boldsymbol{x})\rangle - \tfrac{2(\alpha^\star)^2}{L} \quad = \quad 0,$$

which is equivalent to

$$\alpha^\star \quad = \quad \tfrac{L}{2}\|(\mathbf{H}+\alpha\mathbf{I})^{-1}\nabla f(\boldsymbol{x})\| \quad \overset{(45)}{=} \quad \tfrac{L}{2}\|\boldsymbol{x}^+-\boldsymbol{x}\| \quad \equiv \quad \tfrac{Lr}{2}, \tag{51}$$

where we use $r := \|\boldsymbol{x}^+ - \boldsymbol{x}\|$ as previously. Therefore, from (51) we see that $\alpha^\star$ plays the role of the Cubic Regularization (Nesterov & Polyak, 2006) of our model.

Let us establish the main inequalities on the progress of each step.

**Lemma D.1.** *For the functional residual, it holds*

$$f(\boldsymbol{x}) - f(\boldsymbol{x}^+) \quad \geq \quad \tfrac{L}{3}r^3 + \tfrac{\eta}{2}r^2 \tag{52}$$

*Proof.* Indeed, using Lipschitz continuity of the Hessian and definition of our step, we obtain

$$\begin{aligned}
f(\boldsymbol{x}^+) &\overset{(17),(48)}{\leq} f(\boldsymbol{x}) + \langle\nabla f(\boldsymbol{x}),\boldsymbol{x}^+-\boldsymbol{x}\rangle + \tfrac{1}{2}\langle\nabla^2_+ f(\boldsymbol{x})(\boldsymbol{x}^+-\boldsymbol{x}),\boldsymbol{x}^+-\boldsymbol{x}\rangle - \tfrac{1}{2}\langle\nabla^2_- f(\boldsymbol{x})(\boldsymbol{x}^+-\boldsymbol{x}),\boldsymbol{x}^+-\boldsymbol{x}\rangle + \tfrac{L}{6}r^3 \\
&\leq f(\boldsymbol{x}) + \langle\nabla f(\boldsymbol{x}),\boldsymbol{x}^+-\boldsymbol{x}\rangle + \tfrac{1}{2}\langle\nabla^2_+ f(\boldsymbol{x})(\boldsymbol{x}^+-\boldsymbol{x}),\boldsymbol{x}^+-\boldsymbol{x}\rangle + \tfrac{L}{6}r^3 \\
&\overset{(35)}{=} f(\boldsymbol{x}) - \langle\mathbf{H}(\boldsymbol{x}^+-\boldsymbol{x}),\boldsymbol{x}^+-\boldsymbol{x}\rangle - \alpha r^2 + \tfrac{1}{2}\langle\nabla^2_+ f(\boldsymbol{x})(\boldsymbol{x}^+-\boldsymbol{x}),\boldsymbol{x}^+-\boldsymbol{x}\rangle + \tfrac{L}{6}r^3 \\
&\leq -\alpha r^2 + \tfrac{\sigma^+_\tau+\delta}{2}r^2 + \tfrac{L}{6}r^3.
\end{aligned}$$

Hence, rearranging the terms and using the definition of $\alpha$, we obtain

$$\begin{aligned}
f(\boldsymbol{x}) - f(\boldsymbol{x}^+) &\geq \alpha r^2 - \tfrac{\sigma^+_\tau+\delta}{2}r^2 - \tfrac{L}{6}r^3 \\
&\overset{(46),(51)}{=} \tfrac{L}{2}r^3 + \eta r^2 - \tfrac{\sigma^+_\tau+\delta}{2}r^2 - \tfrac{L}{6}r^3 \\
&\geq \tfrac{L}{3}r^3 + \tfrac{\eta}{2}r^2,
\end{aligned}$$

which is the required bound. □

**Lemma D.2.** *Let $\mathbf{y}^+ := \mathbf{x} + \mathbf{P}(\boldsymbol{x}^+ - \boldsymbol{x})$. Then, we can relate the gradient at $\boldsymbol{y}^+$ and the length $r$ of the step, as follows*

$$\|\nabla f(\boldsymbol{y}^+)\| \quad \leq \quad 2\eta r + Lr^2. \tag{53}$$

*Proof.* By Lipschitz continuity of the Hessian, we have

$$\begin{aligned}
\tfrac{L}{2}r^2 &\geq \tfrac{L}{2}\|\boldsymbol{y}^+-\boldsymbol{x}\|^2 \\
&\overset{(17)}{\geq} \|\nabla f(\boldsymbol{y}^+) - \nabla f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{x})(\boldsymbol{y}^+-\boldsymbol{x})\| \\
&= \|\nabla f(\boldsymbol{y}^+) - \nabla f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{x})\mathbf{P}(\boldsymbol{x}^+-\boldsymbol{x})\| \\
&\overset{(35)}{=} \|\nabla f(\boldsymbol{y}^+) + (\mathbf{H}-\nabla^2 f(\boldsymbol{x}))\mathbf{P}(\boldsymbol{x}^+-\boldsymbol{x}) + \alpha(\boldsymbol{x}^+-\boldsymbol{x})\| \\
&\geq \|\nabla f(\boldsymbol{y}^+)\| - \|(\mathbf{H}-\nabla^2 f(\boldsymbol{x}))\mathbf{P}(\boldsymbol{x}^+-\boldsymbol{x})\| - \alpha r.
\end{aligned}$$

Hence, rearranging the terms, we obtain

$$\begin{aligned}\|\nabla f(\boldsymbol{y}^+)\| &\leq \|(\mathbf{H}-\nabla^2 f(\boldsymbol{x}))\mathbf{P}(\boldsymbol{x}^+-\boldsymbol{x})\| + \alpha r + \tfrac{L}{2}r^2 \\ &\overset{(48)}{=} \|(\mathbf{H}-\nabla^2_+ f(\boldsymbol{x})+\nabla^2_- f(\boldsymbol{x}))\mathbf{P}(\boldsymbol{x}^+-\boldsymbol{x})\| + \alpha r + \tfrac{L}{2}r^2 \\ &\leq 2\eta r + Lr^2,\end{aligned}$$

which is the required bound. □

Combining inequalities (52) and (53) together, we obtain the following bound on the progress.

**Corollary D.3.** *For one step of the method, it holds:*

$$f(\boldsymbol{x}) - f(\boldsymbol{x}^+) \geq \min\Big\{\tfrac{1}{6\sqrt{2L}}\|\nabla f(\boldsymbol{y}^+)\|^{3/2},\ \tfrac{1}{32\eta}\|\nabla f(\boldsymbol{y}^+)\|^2\Big\}. \tag{54}$$

### D.1. Proof of Theorem 5.1

*Proof.* Let us fix some $k \geq 1$ and assume that for any $1 \leq i \leq k$ we have $\|\nabla f(\boldsymbol{y}_i)\| \geq \varepsilon$.

According to (54), we obtain, for $0 \leq i \leq k-1$:

$$\begin{aligned}f(\boldsymbol{x}_i) - f(\boldsymbol{x}_{i+1}) &\geq \min\Big\{\tfrac{1}{6\sqrt{2L}}\|\nabla f(\boldsymbol{y}_{i+1})\|^{3/2},\ \tfrac{1}{32\eta}\|\nabla f(\boldsymbol{y}_{i+1})\|^2\Big\} \\ &\geq \min\Big\{\tfrac{1}{6\sqrt{2L}}\varepsilon^{3/2},\ \tfrac{1}{32\eta}\varepsilon^2\Big\}.\end{aligned}$$

Telescoping this bound for the first $k$ iterations, we get

$$f(\boldsymbol{x}_0) - f^\star \geq f(\boldsymbol{x}_0) - f(\boldsymbol{x}_k) \geq k\min\Big\{\tfrac{1}{6\sqrt{2L}}\varepsilon^{3/2},\ \tfrac{1}{32\eta}\varepsilon^2\Big\},$$

which leads to the required complexity. □

## E. Proofs for Section 6

In this section, we provide a general analysis for the *composite* version of our method (34), when the target objective is convex: $f \in \mathcal{F}_n$ and quasi-Self-Concordant (25) with parameter $M > 0$.

Let us establish the progress for one step of our algorithm $\boldsymbol{x}^+ = \text{CompositeStep}_{\mathbf{H},\alpha}(\boldsymbol{x}, \nabla f(\boldsymbol{x}))$ under this refined smoothness condition.

**Lemma E.1.** *Let $\alpha \geq M\|F'(\boldsymbol{x})\| + \sigma_\tau + \delta$. Then*

$$\langle F'(\boldsymbol{x}^+), \boldsymbol{x} - \boldsymbol{x}^+\rangle \geq \tfrac{1}{2\alpha}\|F'(\boldsymbol{x}^+)\|^2. \tag{55}$$

*Proof.* By using basic properties of quasi-Self-Concordant functions (see Lemma 2.7 in (Doikov, 2023)), we have, for any two points $\boldsymbol{x}, \boldsymbol{x}^+ \in \mathbb{R}^n$:

$$\|\nabla f(\boldsymbol{x}^+) - \nabla f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{x})(\boldsymbol{x}^+ - \boldsymbol{x})\| \leq \tfrac{M}{2}\langle \nabla^2 f(\boldsymbol{x})(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x}\rangle \cdot \varphi(Mr),$$

where $\varphi(t) \overset{\text{def}}{=} \frac{e^t - t - 1}{t^2}$ is a convex monotone univariate function. Let us show how we can control the right hand side. From (37), we have

$$\varphi(Mr) \leq \varphi\big(\tfrac{M\|F'(\boldsymbol{x})\|}{\alpha}\big) \overset{(*)}{\leq} 1,$$

where in $(*)$ we use our choice of the regularization coefficient: $\alpha \geq M\|F'(\boldsymbol{x})\|$.

For the Hessian, we can use

$$\langle \nabla^2 f(\boldsymbol{x})(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x}\rangle \overset{(18)}{\leq} \sigma_\tau r^2 + \langle \mathbf{H}(\boldsymbol{x}^+ - \boldsymbol{x}), \boldsymbol{x}^+ - \boldsymbol{x}\rangle \overset{(38)}{\leq} \sigma_\tau r^2 + r\|F'(\boldsymbol{x})\|$$
$$\overset{(37)}{\leq} (\tfrac{\sigma_\tau}{\alpha} + 1)\cdot r\|F'(\boldsymbol{x})\| \overset{(**)}{\leq} 2r\|F'(\boldsymbol{x})\|,$$

where we used in $(**)$ that $\alpha \geq \sigma_\tau$.

Therefore, we obtain

$$\|\nabla f(\boldsymbol{x}^+) - \nabla f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{x})(\boldsymbol{x}^+ - \boldsymbol{x})\| \leq Mr\|F'(\boldsymbol{x})\|. \tag{56}$$

Thus, using the definition of new subgradient $F'(\boldsymbol{x}^+)$ from the stationary condition (35), we have

$$\begin{aligned}\|F'(\boldsymbol{x}^+) + \alpha(\boldsymbol{x}^+ - \boldsymbol{x})\| &= \|\nabla f(\boldsymbol{x}^+) - \nabla f(\boldsymbol{x}) - \mathbf{H}(\boldsymbol{x}^+ - \boldsymbol{x})\| \\ &\leq \|\nabla f(\boldsymbol{x}^+) - \nabla f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{x})(\boldsymbol{x}^+ - \boldsymbol{x})\| \\ &\quad + r\|\mathbf{H} - \nabla^2 f(\boldsymbol{x})\| \\ &\overset{(56)}{\leq} r(\sigma_\tau + \delta + M\|F'(\boldsymbol{x})\|).\end{aligned}$$

Taking square of both sides and rearranging the terms, we obtain

$$\langle F'(\boldsymbol{x}^+), \boldsymbol{x} - \boldsymbol{x}^+\rangle \geq \tfrac{1}{2\alpha}\|F'(\boldsymbol{x}^+)\|^2 + \tfrac{r^2}{2\alpha}(\alpha^2 - (\sigma_\tau + \delta + M\|\nabla f(\boldsymbol{x})\|)^2).$$

Taking into account or choice of $\alpha$ completes the proof. □

We denote by $D$ the diameter of the initial sublevel set which we assume to be bounded:

$$D := \sup_{\boldsymbol{x}}\Big\{\|\boldsymbol{x} - \boldsymbol{x}^\star\| \,:\, F(\boldsymbol{x}) \leq F(\boldsymbol{x}_0)\Big\} < +\infty.$$

We prove the following result.

**Theorem E.2.** *Let $f \in \mathcal{F}_n$ be convex and quasi-Self-Concordant wtih constant $M$. Consider iterations $\{\boldsymbol{x}_k\}_{k\geq 0}$ of algorithm (34) with*

$$\alpha_k = M\|F'(\boldsymbol{x}_k)\| + \sigma_\tau + \delta,$$

*where $F'(\boldsymbol{x}_k)$ is defined by (36) for $k \geq 1$, and $F'(\boldsymbol{x}_0) := \nabla f(\boldsymbol{x}_0) + \boldsymbol{s}$, for an arbitrary initial subgradient $\boldsymbol{s} \in \partial\psi(\boldsymbol{x}_0)$. Then, for any $\varepsilon > 0$, it is enough to do*

$$K = 4\left\lceil \left(MD + \tfrac{\sigma_\tau + \delta}{2\mu}\right)\ln\tfrac{F(\boldsymbol{x}_0) - F^\star}{\varepsilon} + \ln\tfrac{\|F'(\boldsymbol{x}_0)\|D}{\varepsilon}\right\rceil \tag{57}$$

*steps to ensure $F(\boldsymbol{x}_k) - F^\star \leq \varepsilon$.*

*Proof.* Let us prove the following rate of convergence, for the iterations of our method,

$$F(\boldsymbol{x}_k) - F^\star \leq \exp\Big(-\tfrac{k}{4}\Big[\tfrac{\sigma_\tau + \delta}{2\mu} + MD\Big]^{-1}\Big)\big(F(\boldsymbol{x}_0) - F^\star\big) + \exp\Big(-\tfrac{k}{4}\Big)\cdot\|F'(\boldsymbol{x}_0)\|D, \tag{58}$$

which immediately leads to the complexity bound (57).

We denote $g_k = \|F'(\boldsymbol{x}_k)\|$. Then, by convexity, we have for every iteration $k \geq 0$:

$$\begin{aligned}F(\boldsymbol{x}_k) - F(\boldsymbol{x}_{k+1}) &\geq \langle F'(\boldsymbol{x}_{k+1}), \boldsymbol{x}_k - \boldsymbol{x}_{k+1}\rangle \\ &\overset{(55)}{\geq} \tfrac{1}{2(\sigma_\tau + \delta + Mg_k)}\big(\tfrac{g_{k+1}}{g_k}\big)^2 g_k^2 \\ &= \tfrac{1}{2}\big(\tfrac{g_{k+1}}{g_k}\big)^2\cdot\Big(\tfrac{\sigma_\tau + \delta}{g_k^2} + \tfrac{M}{g_k}\Big)^{-1}.\end{aligned} \tag{59}$$

Denoting the functional residual by $F_k \stackrel{\text{def}}{=} F(\boldsymbol{x}_k) - F^\star$, we have by convexity and strong convexity:

$$g_k \;\geq\; \tfrac{F_k}{D}, \qquad g_k^2 \;\geq\; 2\mu F_k.$$

Substituting these bounds into (59), we obtain

$$F_k - F_{k+1} \;\geq\; \tfrac{1}{2}\big(\tfrac{g_{k+1}}{g_k}\big)^2 \cdot \Big(\tfrac{\sigma_\tau+\delta}{2\mu F_k} + \tfrac{MD}{F_k}\Big)^{-1} \;=\; \tfrac{q}{2}\big(\tfrac{g_{k+1}}{g_k}\big)^2 F_k, \tag{60}$$

where condition number $q$ is defined by

$$q \;:=\; \Big(\tfrac{\sigma_\tau+\delta}{2\mu} + MD\Big)^{-1}.$$

To show the desired rate, we use concavity of logarithm,

$$\ln \tfrac{a}{b} \;=\; \ln a - \ln b \;\geq\; \tfrac{1}{a}(a-b), \qquad \forall a, b > 0,$$

and conclude that

$$\ln \tfrac{F_k}{F_{k+1}} \;\geq\; \tfrac{1}{F_k}(F_k - F_{k+1}) \;\stackrel{(60)}{\geq}\; \tfrac{q}{2}\big(\tfrac{g_{k+1}}{g_k}\big)^2.$$

Telescoping this bound and using inequality between arithmetic and geometric means, we get

$$\begin{aligned}
\ln \tfrac{F_0}{F_k} \;&\geq\; \tfrac{q}{2}\sum_{i=0}^{k-1}\big[\tfrac{g_{i+1}}{g_i}\big]^2 \;\geq\; \tfrac{kq}{2}\Big[\prod_{i=0}^{k-1}\tfrac{g_{i+1}}{g_i}\Big]^{2/k} \;=\; \tfrac{kq}{2}\big[\tfrac{g_k}{g_0}\big]^{2/k} \\
&=\; \tfrac{kq}{2}\exp\big[\tfrac{2}{k}\ln\tfrac{g_k}{g_0}\big] \;\geq\; \tfrac{kq}{2}\Big(1 + \tfrac{2}{k}\ln\tfrac{g_k}{g_0}\Big) \;\geq\; \tfrac{kq}{2}\Big(1 + \tfrac{2}{k}\ln\tfrac{F_k}{g_0 D}\Big).
\end{aligned}$$

The last inequality leads to the required bound (58). □

### E.1. Proof of Theorem 6.1

*Proof.* It follows immediately from Theorem E.2 by substituting the non-composite case $\psi \equiv 0$. □