Iteration Complexity & Accelerated Algorithms

Thomas Zhang

Contents

Abstract

In this article, I aim to survey the important literature surrounding Nesterov's Accelerated Gradient Method and its descendants. I will explore the worst-case objective functions that spawned the need for Nesterov's initial accelerated gradient scheme−namely, attaining the $\Omega\left(1/k^2\right)$ iteration complexity. The middle section of the article will discuss how Nesterov-type acceleration can be applied to algorithms for objective functions that are "composite", in other words the combination of a nice, convex, and smooth function with a convex, *non-smooth*, yet somehow simple function. These objective functions are related to wide classes of real-life applications, including ℓ^1 regularization (a.k.a. LASSO) and low-rank matrix optimization. We will conclude this article with a thorough exploration of a very new paper by Nesterov that considers the case where your objective function is convex (not necessarily strongly) and p-times differentiable, where, among other results, he shows local search tensor methods can attain no more than a $\Omega\left(1/k^{\frac{3p+1}{2}}\right)$ convergence rate. Note that plugging in $p = 1$ gets us the same bounds as in the gradient case.

Disclaimer

For all the bounds that I prove here, I am considering the case where the objective is possibly not strongly convex, which means all the convergence rates are going to be sub-linear (sub-exponential). However, the examples of Nesterov acceleration shown below all also accelerate on strongly convex objectives, where the convergence is usually improved by a square factor, somewhat analogous to the $\Omega(1/k) \to \Omega(1/k^2)$ improvement we see in Section 1, and the proof structures are all essentially the same. Therefore, to spare my poor, winter-chilled hands, I will only report the results assuming we might not have strong convexity.

1 Nesterov's Accelerated Gradient Method

If we were to provide a narrative to give some motivation for all this accelerated gradient method business, let us start with the scene before Nesterov comes in. Relatively soon after gradient method was first phrased in modern optimization terms, people knew that for objective functions that are smooth and strongly convex, many gradient methods enjoyed a linear convergence, and there were examples to show that this bound was tight. However, what happened if strong convexity were taken away was not as cut-and-dry of a problem. A lot of examples showed that many of the existing gradient methods slowed down and could only attain a $O(1/k)$ sub-linear convergence. This is where Nesterov first comes in. The following section details a easier version of Nesterov's original "worst-case" function and the method he describes that attains that bound.

1.1 "Worst Function in the World"

Before we can discuss the accelerated algorithm, we have to understand why it was considered "optimal". We will walk through the example that Nesterov proposes in [6]. We characterize first-order local search methods in the following way:

Definition 1.1 *The class $\mathcal{M}$ contains methods that generate sequences of points $\{x_k\}$ that obey*

$$x_k \in x_0 + \textbf{\textit{span}}\left(\nabla f(x_0), \nabla f(x_1), \ldots, \nabla f(x_{k-1})\right). \tag{1.1}$$

Notice that this includes the steepest descent method, as well as many other popular methods. We now propose a family of functions that will contain the "worst function in the world".

Definition 1.2 *Consider the family of functions $\mathcal{F} = \{f_k\}_{k=1}^n$, $f_k : \mathbb{R}^n \to \mathbb{R}$. Fix some $L > 0$.*

$$\begin{aligned} f_k(x) &= \frac{L}{4}\left(\frac{1}{2}\left[x_1^2 + \sum_{i=1}^{k-1}(x_i - x_{i+1})^2 + x_k^2\right] - x_1\right) \\ &= \frac{L}{4}\left(\frac{1}{2}x^\top A_k x - e_1^\top x\right), \end{aligned}$$

where e_1 is the first coordinate unit basis vector, and A_k is the following tri-diagonal block matrix

$$A_k = \begin{bmatrix} 2 & -1 & 0 & & \\ -1 & 2 & -1 & 0 & \\ 0 & -1 & 2 & & \\ & & & & \\ & 0 & & 0 & \\ & & & & \end{bmatrix}.$$

We observe that since f_k are quadratic functions, we immediately have the Hessian

$$\nabla^2 f(x_k) = \frac{L}{4} A_k$$

and from our definition above, we find that $\nabla^2 f_k(x)$ is positive semi-definite (and strictly so for $k < n$), and

$$0 \leq \nabla^2 f(x) \leq L I_n.$$

Therefore, for $k < n$, we have that f_k are convex (but not strictly convex) and L-smooth. We can also explicitly compute the optimal value for any given f_k. We simply solve the equation

$$\begin{aligned}
\nabla f_k(x) &= \frac{L}{4}\left(A_k x - e_1\right) = 0 \\
\implies x_i^* &= \begin{cases} 1 - \frac{i}{k+1}, & i \leq k \\ 0, & i > k \end{cases} \\
\implies f_k^* &= \frac{L}{4}\left\{\frac{1}{2}\left[x_1^{*2} + \sum_{i=1}^{k-1}(x_i^* - x_{i+1}^*)^2 + x_k^{*2}\right] - x_1^*\right\} \\
&= \frac{L}{4}\left(\frac{1}{2} x^{*\top} A_k x^* - e_1^\top x^*\right) \\
&= \frac{L}{4}\left(-\frac{1}{2} e_1^\top x^*\right) \\
&= -\frac{L}{8} e_1^\top x^* \\
&= -\frac{L}{8}\left(1 - \frac{1}{k+1}\right). \qquad (1)
\end{aligned}$$

Furthermore, we can make a rough upper bound on $\|x^*\|_2$:

$$\begin{aligned}
\|x^*\|_2^2 &= \sum_{i=1}^{n} (x_i^*)^2 \\
&= \sum_{i=1}^{k} \left(1 - \frac{2i}{k+1} + \left(\frac{i}{k+1}\right)^2\right) \\
&= k - \frac{2}{k+1}\sum_{i=1}^{k} i + \frac{1}{(k+1)^2}\sum_{i=1}^{k} i^2 \\
&= k - \frac{2}{k+1}\frac{k(k+1)}{2} + \frac{1}{(k+1)^2}\frac{k(k+1)(2k+1)}{6} \\
&= k - k + \frac{1}{k+1}\frac{k(2k+1)}{6} \\
&\leq \frac{1}{k+1}\frac{(k+1)(2k+2)}{6} \\
&= \frac{1}{3}(k+1). \qquad (2)
\end{aligned}$$

We now make some subspace arguments. We denote $\mathbb{R}^n_{[k]} := \mathbf{span}\,(e_1, \ldots, e_k)$. Observe that for $x \in \mathbb{R}^n_{[k]}$, we have that $f_p(x) = f_k(x)$ for all $p > k$, which is clearly true by the function's definition from Definition 1.2. We now prove a truly key lemma that tells us how iterates from first order methods from class $\mathcal{M}$ behave:

Lemma 1.3 (Subspace Expansion of Gradient Methods) *Fix some $p \in [n]$ and set $x_0 = 0$. We recursively define a sequence $\{x_k\}_{k=0}^p$, where*

$$x_k \in S_k := \boldsymbol{span}\left(\nabla f_p(x_0), \nabla f_p(x_1), \ldots, \nabla f_p(x_{k-1})\right).$$

Then, $S_k \subseteq \mathbb{R}^n_{[k]}$.

Proof of lemma: This follows from a simple induction argument. For our base case, observe that since $x_0 = 0$, we have $\nabla f_p(x_0) = 0 - \frac{L}{4}e_1 = -\frac{L}{4}e_1$, and the span S_1 is clearly $\mathbb{R}^n_1$. For the induction step, assume $S_k \subseteq \mathbb{R}^n_{[k]}$. Take any $x \in \mathbb{R}^n_{[k]}$. Observe that $\nabla f_p(x) = \frac{L}{4}\left(A_p x - e_1\right)$, and A_p is a tri-diagonal matrix, so $A_p x \in \mathbb{R}^n_{k+1}$. Therefore, we have shown by induction that $S_{k+1} \subseteq \mathbb{R}^n_{k+1}$, which completes the proof of the lemma.

In my opinion, even if this is one of the technically simpler lemmas, it is in many ways the key phenomenon that allows the "worst function in the world" to exist. We saw from our construction of the family of quadratic functions that the optimal value of a given f_k is inherently related to the dimension of the domain subspace: x_k^* is supported only on the first k coordinates. Therefore, by showing that gradient methods cannot explore more than one additional dimension per iteration, we have very succinctly tied down the convergence rate of gradient methods to the slow decrease of the optimal value with respect to dimension of the domain. By removing strong convexity we (seem to) admit functions where exploring one dimension at a time won't get us linear convergence. We will see a more intricate version of

this argument when we consider higher-order tensor methods in Section 3, but the conclusion will be the same: the dimension of the search space can only increase one dimension per iteration. We are now ready to propose the worst function in the world.

Theorem 1.4 *Let us fix $k \leq \frac{1}{2}(n-1)$ such that $2k+1 \leq n$. We claim that there is some f_p that fulfills the following property:*

$$\begin{aligned} f_p(x_k) - f_p^* &\geq \frac{3}{32} L \left\| x_0 - x^* \right\|^2 \frac{1}{(k+1)^2} \\ &= \Omega\left(\frac{1}{k^2}\right). \end{aligned}$$

Proof of theorem: First we observe that it is sufficient to assume $x_0 = 0$, because we can simply consider the centered function $\overline{f}_p(x_k) = f_p(x_k + x_0)$ and use any gradient method from $\mathcal{M}$ on that. Let us set $p = 2k+1 \leq n$. We use (1) and (2) to bound the following quantity:

$$\begin{aligned} \frac{f_p(x_k) - f_p^*}{\left\| x_0 - x^* \right\|^2} &\geq \frac{-\frac{L}{8}\left(1 - \frac{1}{k+1} - 1 + \frac{1}{2k+1}\right)}{\frac{1}{3}(2k+2)} \\ &\geq \frac{3}{16} L \frac{\frac{1}{2}\frac{1}{k+1}}{\frac{1}{k+1}} \\ &= \frac{3}{32} L \frac{1}{(k+1)^2} \\ f_p(x_k) - f_p^* &= \frac{3}{32} L \left\| x_0 - x^* \right\|^2 \frac{1}{(k+1)^2}. \end{aligned}$$

This completes the proof of the main theorem.

Thus, we have just shown that if we consider minimizing convex (but not strictly), smooth functions, gradient methods cannot in general attain a faster convergence to the optimal than $\Omega\left(\frac{1}{k^2}\right)$. Nesterov then showed that taking our favorite steepest descent method, we only actually enjoy $\Omega\left(\frac{1}{k}\right)$ convergence [6], which would tell us either the example isn't not as bad as it can get, or that the methods themselves can be improved. Nesterov's accelerated gradient method resolves this question.

1.2 The Algorithm and Analysis

Most of the following derivations are adapted and regurgitated from [5] and [6].

Nesterov's accelerated gradient method can be seen as a funky, modified proximal gradient method, where instead of the prox operator depending on the previous iterate location, it depends on a combination of the previous two points. I can't say much on the philosophical reason why this works, but on the algebraic side, we will see the particular choice of step size is the solution to a quadratic recurrence relation. Let us propose a de-clunked version of the algorithm: given a convex, L-smooth objective function $f(x)$ on domain X

Algorithm 1 Nesterov's Accelerated Gradient Method

1: Initialize $x_0 = y_1 \in X$ arbitrarily.
2: **for** $k = 1, 2, \ldots$ **do**
3: $\quad x_k = \arg\min_{x \in X} f(y_k) + \nabla f(y_k)^\top (x - y_k) + \frac{L}{2} \|x - y_k\|_2^2$
4: $\quad y_{k+1} = x_k + \frac{t_k - 1}{t_{k+1}}(x_k - x_{k-1})$
5: **end for**
6: **return** $x_k, f(x_k)$

We observe that line 3 rings familiar of the vanilla proximal gradient method, where y_k would have been replaced with x_{k-1}. As for the constant t_k controlling the step size, we (interestingly) generate it through the recurrence relation

$$\begin{aligned} t_1 &= 1 \\ t_k^2 - t_k &= t_{k-1}^2 \\ t_{k+1} &= \frac{1 + \sqrt{4t_k^2 + 1}}{2}. \end{aligned}$$

For the main result, we want to show the above algorithm has the following convergence rate:

Theorem 1.5 *Nesterov's accelerated gradient method has a convergence rate of*

$$f(x_k) - f(x^*) \leq \frac{f(x_0) - f(x^*) + \frac{L\|x_0 - x^*\|_2^2}{2}}{t_k^2} \tag{1.2}$$

$$\leq \frac{4(f(x_0) - f(x^*)) + 2L\|x_0 - x^*\|_2^2}{(k+1)^2}. \tag{1.3}$$

We will break up the computation-heavy proof into a few lemmas:

Lemma 1.6 *Given Nesterov constant t_k, we have the following estimate*

$$t_k \geq \frac{k+1}{2}. \tag{1.4}$$

Proof of lemma: we prove this by induction. Base case $t_1 = 1 \geq \frac{1+1}{2}$. For the induction step, assume that $t_k \geq \frac{k+1}{2}$. Then we have from the definition of t_{k+1}:

$$\begin{aligned} t_{k+1} &= \frac{1 + \sqrt{4t_k^2 + 1}}{2} \\ &\geq \frac{1 + \sqrt{4\frac{(k+1)^2}{4} + 1}}{2} \\ &\geq \frac{1 + \sqrt{(k+1)^2}}{2} \\ &= \frac{k+2}{2}. \end{aligned}$$

This completes the proof by induction.

Lemma 1.7 *Let j be an iteration of the algorithm. We claim*

$$f(x_{j+1}) \leq f(x) + L(x_{j+1} - y_{j+1})^\top (x - x_{j+1}) + \frac{L}{2} \|x_{j+1} - y_{j+1}\|_2^2, \tag{1.5}$$

for all $x \in X$.

Lemma 1.8 *Let j be an iteration of the algorithm. From Lemma 1.7, we can get*

$$\begin{aligned} & t_{j+1}^2 \left(f(x_{j+1}) - f(x^*)\right) - t_j^2 \left(f(x_j) - f(x^*)\right) \\ \leq & \frac{L}{2} \left(\|t_j x_j - (t_j - 1)x_{j-1} - x^*\|_2^2 - \|t_{j+1} x_{j+1} - (t_{j+1} - 1)x_j - x^*\|_2^2\right). \end{aligned} \tag{1.6}$$

Proof of Theorem 1.5 from Lemma 1.8: this follows by observing that summing inequality (1.6) over j, we get a telescoping series on both ends, and some light rearranging afterward gets us what we want. Let's say we iterate from 0 to $k-1$:

$$\begin{aligned} & \sum_{j=0}^{k-1} \left(t_{j+1}^2 \left(f(x_{j+1}) - f(x^*)\right) - t_j^2 \left(f(x_j) - f(x^*)\right)\right) \\ & = t_k^2 (f(x_k) - f(x^*)) - (f(x_0) - f(x^*)) \\ & \sum_{j=0}^{k-1} \left[\frac{L}{2} \left(\|t_j x_j - (t_j - 1)x_{j-1} - x^*\|_2^2 - \|t_{j+1} x_{j+1} - (t_{j+1} - 1)x_j - x^*\|_2^2\right)\right] \\ & = \frac{L}{2} \left[\|x_0 - x^*\|_2^2 - \|t_k x_k - (t_k - 1)x_{k-1} - x^*\|_2^2\right] \end{aligned}$$

Putting the above into the respective places in the inequality, we get

$$\begin{aligned} t_k^2 (f(x_k) - f(x^*)) - (f(x_0) - f(x^*)) &\leq \frac{L}{2} \left[\|x_0 - x^*\|_2^2 - \|t_k x_k - (t_k - 1)x_{k-1} - x^*\|_2^2\right] \\ &\leq \frac{L}{2} \|x_0 - x^*\|_2^2 \\ f(x_k) - f(x^*) &\leq \frac{f(x_0) - f(x^*) + \frac{L\|x_0 - x^*\|_2^2}{2}}{t_k^2} \\ &\leq \frac{4(f(x_0) - f(x^*)) + 2L\|x_0 - x^*\|_2^2}{(k+1)^2} \text{ using Lemma 1.6.} \end{aligned}$$

We must now prove Lemmas 1.7 and 1.8.

Proof of Lemma 1.7: since we have by definition

$$x_{j+1} = \arg\min_{x \in X} f(y_{j+1}) + \nabla f(y_{j+1})^\top (x - y_{j+1}) + \frac{L}{2} \|x - y_{j+1}\|^2 := \arg\min_{x \in X} g(x).$$

Notice that the left hand side is a convex function and x_{j+1} by definition is an optimal solution to it, so by the first order optimality condition we have that

$$\begin{aligned} \nabla g(x_{j+1})^\top (x - x_{j+1}) &\geq 0 \\ \left(\nabla f(y_{j+1}) + L(x_{j+1} - y_{j+1})\right)^\top (x - x_{j+1}) &\geq 0 \quad \forall x \in X. \end{aligned} \tag{1.7}$$

From the convexity and L-smoothness of f, we also have that

$$f(x) \leq f(y) + \nabla f(y)^\top (x - y) + \frac{L}{2} \|x - y\|^2 \quad \forall x, y \in X$$

which, substituting $x = x_{j+1}, y = y_{j+1}$ gets us

$$f(x_{j+1}) \leq f(y_{j+1}) + \nabla f(y_{j+1})^\top (x_{j+1} - y_{j+1}) + \frac{L}{2} \|x_{j+1} - y_{j+1}\|^2 . \tag{1.8}$$

To transfer (1.8) to something that is true for all x, we observe that by convexity of f, we have

$$f(y_{j+1}) \leq f(x) - \nabla f(y_{j+1})^\top (x - y_{j+1}) \quad \forall x \in X. \tag{1.9}$$

Substituting (1.9) into (1.8), we get

$$f(x_{j+1}) \leq f(x) + \nabla f(y_{j+1})^\top (x_{j+1} - x) + \frac{L}{2} \|x_{j+1} - y_{j+1}\|^2 \quad \forall x \in X.$$

Combining the above with (7) we get

$$f(x_{j+1}) \leq f(x) + L(x_{j+1} - y_{j+1})^\top (x - x_{j+1}) + \frac{L}{2} \|x_{j+1} - y_{j+1}\|^2 \quad \forall x \in X.$$

This completes the proof of Lemma 1.7.

Proof of Lemma 1.8: we use Lemma 1.7 and substitute $x = x^*$, $x = x_j$ to get the following two inequalities

$$f(x_{j+1}) \leq f(x^*) + L(x_{j+1} - y_{j+1})^\top (x^* - x_{j+1}) + \frac{L}{2} \|x_{j+1} - y_{j+1}\|^2 \tag{1.10}$$

$$f(x_{j+1}) \leq f(x_j) + L(x_{j+1} - y_{j+1})^\top (x_j - x_{j+1}) + \frac{L}{2} \|x_{j+1} - y_{j+1}\|^2 . \tag{1.11}$$

Now begins a series of awful(ly) technical calculations. Multiply (1.10) by $\frac{1}{t_{j+1}}$ and (1.11) by $1 - \frac{1}{t_{j+1}}$ and then add up the two inequalities with some re-arrangement to get

$$\frac{1}{t_{j+1}}(f(x_{j+1}) - f(x^*)) + \left(1 - \frac{1}{t_{j+1}}\right)(f(x_{j+1}) - f(x_j)) \tag{LHS}$$

$$\leq L(x_{j+1} - y_{j+1})^\top \left(\frac{1}{t_{j+1}} x^* + \left(1 - \frac{1}{t_{j+1}}\right) x_j - x_{j+1}\right) + \frac{L}{2} \|x_{j+1} - y_{j+1}\|^2 \tag{RHS}$$

Cleaning up the left-hand side, we get

$$f(x_{j+1}) - f(x^*) - \left(1 - \frac{1}{t_{j+1}}\right)(f(x_j) - f(x^*)). \tag{LHS}$$

For (RHS), we use the following fact about inner product spaces (polarization identity)

$$(v_1 - v_2)^\top (v_3 - v_1) = \frac{1}{2}\left(\|v_2 - v_3\|^2 - \|v_1 - v_2\|^2 - \|v_1 - v_3\|^2\right) \tag{1.12}$$

to get

$$\frac{L}{2}\left(\left\|y_{j+1}-\left(1-\frac{1}{t_{j+1}}\right)x_j-\frac{1}{t_{j+1}}x^*\right\|^2-\left\|x_{j+1}-\left(1-\frac{1}{t_{j+1}}\right)x_j-\frac{1}{t_{j+1}}x^*\right\|^2\right)$$
$$=\frac{L}{2t_{j+1}^2}\left(\|t_{j+1}y_{j+1}-(t_{j+1}-1)x_j-x^*\|^2-\|t_{j+1}-(t_{j+1}-1)x_j-x^*\|^2\right). \qquad \text{(RHS)}$$

For the coup de grace, we will see exactly why the particular choice of Nesterov's constant is important: recall that we set

$$y_{j+1}=x_j+\frac{t_j-1}{t_j+1}(x_j-x_{j-1})$$
$$\iff t_{j+1}y_{j+1}-(t_{j+1}-1)x_j=t_jx_j-(t_j-1)x_{j-1}.$$

Also recall that we generated constants t_j such that $t_{j+1}(t_{j+1}-1)=t_j^2$. Let's multiply LHS and RHS by t_{j+1}^2 to get

$$t_{j+1}^2(f(x_{j+1})-f(x^*))-\left(t_{j+1}^2-t_{j+1}\right)(f(x_j)-f(x^*)) \qquad \text{(LHS)}$$
$$\leq\frac{L}{2}\left(\|t_{j+1}y_{j+1}-(t_{j+1}-1)x_j-x^*\|^2-\|t_{j+1}-(t_{j+1}-1)x_j-x^*\|^2\right) \qquad \text{(RHS)}$$
$$\iff t_{j+1}^2(f(x_{j+1})-f(x^*))-t_j^2(f(x_j)-f(x^*))$$
$$\leq\frac{L}{2}\left(\|t_jx_j-(t_j-1)x_{j-1}-x^*\|^2-\|t_{j+1}x_{j+1}-(t_{j+1}-1)x_j-x^*\|^2\right),$$

which completes the proof of Lemma 1.8. From our earlier discussion, this completes the proof of the main theorem.

2 Composite Objective Functions

2.1 Prelude: Optimization is Hard

Let's say we were particularly ambitious and wanted to tackle the problem of unconstrained nonconvex nonsmooth minimization. Intuitively, this should not be possible, because if we look at the broad class of simple combinatorial problems that are NP-hard, many of them can actually be seen as a continuous optimization problem whose global optima are provably difficult to calculate. I particularly like this reduction to Boolean Knapsack by Nesterov [7] as a pedagogical example. In this example, Nesterov shows that not only is finding the global optima of a nonconvex nonsmooth function NP-hard, simply finding a *descent direction* for such a function is also NP-hard. In other words, nonconvex nonsmooth functions can be so misbehaved that even deciding whether local improvement is possible is intractable. Wow that's a shame!

Definition 2.1 *Let us define the following function* $f:\mathbb{R}^n\to\mathbb{R}$

$$f(x)=\left(1-\frac{1}{\gamma}\right)\max_i|x_i|-\min_i|x_i|+\left|c^\top x\right|.$$

where c is any positive integer vector $c \in \mathbb{Z}^n_{++}$ and $\gamma = \sum_i c_i$. We observe that f is a piece-wise linear function, and $f(0) = 0$.

We now have the following sad result. Let's say we're sitting at 0, and we want to decide if we can decrease the function value:

Theorem 2.2 (Descending is Hard) *Finding any $x \in \mathbb{R}^n$ such that $f(x) < 0$ is NP-hard.*

Proof of Theorem 2.2: Let us consider the following problem in terms of Boolean Knapsack: given a sequence of positive integers $\{c_1, \ldots, c_n\}$, we want to know if there is an assignment of signs $\sigma \in \{\pm 1\}^n$ such that

$$\vec{c}^\top \sigma = \sum_{i=1}^{n} c_i \sigma_i = 0.$$

This problem is NP-complete. Keeping this in mind, we return to the function f: we first show that f is positively homogenous (i.e. can pull out positive scalars).

Lemma 2.3 *Let $ax = x'$, where $a > 0$. Then,*

$$f(x') = f(ax) = af(x).$$

Proof of lemma: straighforward calculation

$$\begin{aligned} f(ax) &= \left(1 - \frac{1}{\gamma}\right) \max_i |ax_i| - \min_i |ax_i| + \left|c^\top (ax)\right| \\ &= \left(1 - \frac{1}{\gamma}\right) \max_i a\,|x_i| - \min_i a\,|x_i| + a\left|c^\top x\right| \\ &= a\left[\left(1 - \frac{1}{\gamma}\right) \max_i |x_i| - \min_i |x_i| + \left|c^\top x\right|\right] \\ &= af(x). \end{aligned}$$

Using this lemma, if we find an x' such that $f(x') < 0$, then we can find an appropriate scaling to get x where $f(x) < 0$ and $\max_i x_i = 1$. Setting $\delta = \left|c^\top x\right|$, we now have

$$\begin{aligned} f(x) &= \left(1 - \frac{1}{\gamma}\right) - \min_i |x_i| + \delta < 0 \\ \implies \quad \min_i |x_i| &> \left(1 - \frac{1}{\gamma}\right) + \delta \\ |x_i| &> \left(1 - \frac{1}{\gamma}\right) + \delta, \quad \text{for all } i. \end{aligned}$$

Let us define vector σ such that $\sigma_i = \operatorname{sign}(x'_i)$ so that we get

$$\sigma_i x_i > \left(1 - \frac{1}{\gamma}\right) + \delta, \quad \text{for all } i.$$

We observe that

$$|\sigma_i - x_i| = 1 - \sigma_i x_i$$

and combining that with the above inequality we get

$$\max_i |\sigma_i - x_i| < \frac{1}{\gamma} - \delta.$$

Using the triangle inequality, we now split open

$$\begin{aligned}
\left|c^\top \sigma\right| &\leq \left|c^\top x\right| + \left|c^\top (\sigma - x)\right| \\
&\leq \delta + \sum_{i=1}^{n} c_i \max_i |\sigma_i - x_i| \\
&< \delta + \gamma \left(\frac{1}{\gamma} - \delta\right) \quad \text{recalling } \gamma := \sum_i c_i \\
&= \delta(1-\gamma) + 1 \\
&\leq 1 \quad \text{since } \gamma \geq 1.
\end{aligned}$$

We observe that since $\left|c^\top \sigma\right|$ is a non-negative value and c and σ are both positive integer vectors, the only way for $\left|c^\top \sigma\right| < 1$ is for $c^\top \sigma = 0$. However, we should realize that this solves the Boolean Knapsack problem. Since $\sigma = \text{sign}\,(x')$, this tells us that finding an x' such that $f(x') < 0$ in the first place must have been hard.

2.2 The Problem and ISTA

From the previous section, if we are to hope for efficient solutions, we must scale back our ambitions a bit and restrict our attention to a certain class of important functions: composite functions. Composite objective functions come in the form

$$\phi(x) = f(x) + g(x)$$

where f is a convex, smooth function, and h is a convex, but non-smooth function. Usually, we still want h to be some sort of "simple", well-understood function. Observe that this is a more general class of functions that the ones considered in Section 1. One might wonder why this is a useful class of functions to optimize. An classic example is the LASSO problem in regression, where we might have a problem that looks like

$$\min_x \|Ax - b\|_2^2 + \gamma \|x\|_1$$

where we introduce an ℓ^1 regulariztion term to promote the sparsity of the optimal solution (why ℓ^1 induces sparsity is a whole other rabbit hole; see [4, 11]). This is a problem that people would like to solve quickly and at a large-scale; in fact, it might be argued that these sorts of problems are why composite functions became of interest in the first place. A natural place to look toward for dimension-independent algorithms would be gradient methods. Now the question becomes: is there a smart way to make use of the special structure of the problem

to first come up with an algorithm that performs well, and then prove that it does−rather than blindly applying subgradient methods on $\phi = f + g$ together, which is a non-smooth optimization.

It can be relatively easily shown that subgradient methods with Polyak step sizes attains a convergence rate of

$$\phi(x_k) - \phi^* = \Omega(1/\sqrt{k})$$

on general convex non-smooth objective functions. We will show that a slightly different, but simple algorithm will allow us to improve the convergence analysis to get

$$\phi(x_k) - \phi^* = \Omega(1/k),$$

which we note matches the convergence guarantee for steepest descent on a general convex smooth objectives. The following algorithm is known as the Iterative Shrinkage-Thresholding Algorithm (ISTA), and is really an adaptation of the proximal point algorithm [10]: given objective $f(x) + g(x)$ where ∇f has Lipschitz constant L,

Algorithm 2 Iterative Shrinkage-Thresholding Algorithm

1: Initialize $x_0 \in X$ arbitrarily.
2: **for** $k = 1, 2, \ldots$ **do**
3: $\quad x_{k+1} := \arg\min_{x \in X} f(x_k) + \nabla f(x_k)^\top (x - x_k) + \frac{L}{2} \|x - x_k\|_2^2 + g(x)$
4: **end for**
5: **return** $x_k, f(x_k)$

Before proving the convergence result, we note the following: the minimization subroutine is simply a first-order approximation of f along with g as extra baggage. Therefore, an implicit assumption that ISTA makes is that there is some structure in g that we can leverage that makes the minimization subroutine efficient. Fortunately for us, for many of the applications that we care about, this is true (ℓ^1 regularization [1], low-rank matrix optimization [2, 9], compressed sensing [4]). Let us now prove the convergence result.

Theorem 2.4 (ISTA Convergence) *For convex objective functions $\phi(x) = f(x) + g(x)$, where f is convex and L-smooth, and g is convex but not necessarily smooth, then ISTA attains the following convergence guarantee*

$$\phi(x_k) - \phi(x^*) \leq \frac{L \|x_0 - x^*\|_2^2}{2k}.$$

Proof of Theorem 2.4: let us abuse notation a little bit and denote $g'(x) \in \partial g(x)$ a subgradient of g at x. The proof structure will be strikingly similar to the proof for Nesterov's accelerated algorithm. Consider iteration t. Looking at the minimization subroutine, the first-order optimality condition at x_{t+1} gets us

$$\left[\nabla f(x_t) + L(x_{t+1} - x_t) + g'(x_{t+1})\right]^\top (x - x_{t+1}) \geq 0 \quad \forall x \in X. \tag{2.1}$$

Now, from the convexity of g we also have

$$\begin{aligned} g(x) - g(x_{t+1}) &\geq g'(x_{t+1})^\top (x - x_{t+1}) \\ g(x_{t+1}) &\leq g(x) - g'(x_{t+1})^\top (x - x_{t+1}) \\ &\leq g(x) + \left[\nabla f(x_t) + L(x_{t+1} - x_t)\right]^\top (x - x_{t+1}) \quad \text{from 2.1.} \end{aligned} \tag{2.2}$$

From the fact that x_{t+1} is chosen optimally with respect to the subroutine

$$x_{t+1} = \arg\min_{x \in X} f(x_t) + \nabla f(x_t)^\top (x - x_t) + \frac{L}{2} \|x - x_t\|_2^2$$

and the convexity of f, we have

$$\begin{aligned} f(x_{t+1}) &\leq f(x_t) + \nabla f(x_t)^\top (x_{t+1} - x_t) + \frac{L}{2} \|x_{t+1} - x_t\|_2^2 \\ &\leq f(x) - \nabla f(x_t)^\top (x - x_t) + \nabla f(x_t)^\top (x_{t+1} - x_t) + \frac{L}{2} \|x_{t+1} - x_t\|_2^2 \\ &= f(x) - \nabla f(x_t)^\top (x - x_{t+1}) + \frac{L}{2} \|x_{t+1} - x_t\|_2^2 . \end{aligned} \tag{2.3}$$

Putting 2.2 and 2.3 together, and using the polarization identity 1.12, we get

$$\begin{aligned} f(x_{t+1}) + g(x_{t+1}) &\leq f(x) + g(x) + \frac{L}{2} \left[2(x_{t+1} - x_t)^\top (x - x_{t+1}) - \|x_{t+1} - x_t\|_2^2\right] \\ &= f(x) + g(x) + \frac{L}{2} \left[\|x_t - x\|_2^2 - \|x_{t+1} - x\|_2^2\right] . \end{aligned} \tag{2.4}$$

Setting $x = x^*$, we get

$$\phi(x_{t+1}) \leq \phi(x^*) + \frac{L}{2} \left[\|x_t - x^*\|_2^2 - \|x_{t+1} - x^*\|_2^2\right] .$$

That minus sign hints at a telescoping series argument. We observe that ISTA produces monotonically improving iterates: $\phi(x_{t+1}) \leq \phi(x_t)$. This follows by construction (if one is familiar with the "prox" operator, that is a key reason why proximal methods are powerful). Therefore, certainly the k-th iterate error should be better than the average of the previous iterates

$$\begin{aligned} \phi(x_k) - \phi(x^*) &\leq \frac{\sum_{t=0}^{k-1} \phi(x_{t+1}) - \phi(x^*)}{k} \\ &\leq \frac{L}{2k} \sum_{t=0}^{k-1} \left(\|x_t - x\|_2^2 - \|x_{t+1} - x\|_2^2\right) \\ &\leq \frac{L}{2k} \left(\|x_0 - x\|_2^2 - \|x_k - x\|_2^2\right) \\ &\leq \frac{L \|x_0 - x\|_2^2}{2k}, \end{aligned}$$

which completes the proof. Thus, we have demonstrated that even on not-quite-smooth objectives, we can find a reasonably simple algorithm that attains the same convergence rate as steepest descent methods on smooth convex objectives. This can be seen as the first stage of "acceleration". However, in the next section, we can basically take the machinery introduced in Section 1 to achieve Nesterov acceleration on ISTA.

2.3 A Natural Acceleration: FISTA

The idea behind accelerating ISTA to get FISTA (Fast ISTA) [1] is extremely simple in concept: on smooth convex objectives, we saw that messing around with the step size in the vanilla steepest descent with exact line search gets us an accelerated gradient method. FISTA is nothing more than ISTA with the Nesterov step size:

Algorithm 3 Fast Iterative Shrinkage-Thresholding Algorithm

1: Initialize $x_1 = y_0 \in X$ arbitrarily.
2: **for** $k = 1, 2, \ldots$ **do**
3: $\quad x_k := \arg\min_{x \in X} f(y_k) + \nabla f(y_k)^\top (x - y_k) + \frac{L}{2} \|x - y_k\|_2^2 + g(x)$
4: $\quad y_{k+1} = x_k + \frac{t_k - 1}{t_{k+1}} (x_k - x_{k-1})$
5: **end for**
6: **return** $x_k, f(x_k)$

where $\{t_k\}$ are generated by the same recurrence relation as in Algorithm 1.2:

$$\begin{aligned} t_1 &= 1 \\ t_k^2 - t_k &= t_{k-1}^2 \\ t_{k+1} &= \frac{1 + \sqrt{4t_k^2 + 1}}{2}. \end{aligned}$$

Theorem 2.5 (FISTA Convergence) *On convex objective $\phi(x) = f(x) + g(x)$ where f is convex and L-smooth, and g is convex and possibly non-smooth, we have a convergence rate*

$$\phi(x_k) - \phi(x^*) \leq \frac{4\left[\phi(x_1) - \phi(x^*)\right] + 2L\|x_1 - x^*\|_2^2}{(k+1)^2} = \Omega\left(1/k^2\right)$$

What I think is extra nice about this acceleration scheme is that the proof of the convergence rate is actually almost identical to the one proposed in Section 1, except carrying along the non-smooth part $g(x)$. The crux of the proof is the analog of Lemma 1.7:

$$\phi(x_{j+1}) \leq \phi(x) + L(x_{j+1} - y_{j+1})^\top (x - x_{j+1}) + \frac{L}{2}\|x_{j+1} - y_{j+1}\|_2^2 \quad \forall x \in X,$$

which itself follows from similar convex analysis as in Section 1. The rest of the proof is exactly the same as the one seen in Section 1.

Nice as this acceleration is, we remark that this is not in fact the acceleration scheme that Nesterov himself proposes in [7], which is somewhat more general in its structure. If one is wondering "if we can already achieve $\Omega(1/k^2)$ for composite objectives, what prevents us from isolating a smooth component of any convex objective and using FISTA?" This lies in the implicit assumption that the proximal mapping subroutine can be efficiently solved; if it can't, then each iteration may take an immense amount of time. This is why we insist that the non-smooth part of the objective function must be "simple", i.e. have structure that allows for fast optimization.

3 Nesterov Acceleration for Higher-Order Methods

Let us consider a similar scenario to the one faced in Section 1. We have an objective function that is convex, but not strongly, and p-times differentiable. Let us introduce some notation. We will work with tensors, where we have the following notation for directional derivatives:

$$D^p f(x)[h_1, \ldots, h_p]$$

is the directional derivative of f along $h_1, \ldots, h_p$. From this, we can define the norm

$$\|D^p f(x)\| = \max_{h_1,\ldots,h_p} \{D^p f(x)[h_1, \ldots, h_p] : \|h_i\| \leq 1\}.$$

We note that analogously to the Hessian, the norm is (in general) attained when $h_1 = \cdots = h_p$:

$$\|D^p f(x)\| = \max_h \{|D^p f(x)[h]^p| : \|h\| \leq 1\}$$

Given convex f that is p times differentiable, we write its Taylor expansion at y

$$\begin{aligned} f(x) &= f(x) + \sum_{j=1}^{p} \frac{1}{j!} D^j f(x)[y-x]^j + o(\|h\|^p) \\ &:= \Phi_{x,p}(y) + o(\|h\|^p) \end{aligned}$$

in particular

$$f(x+h) = f(x) + \sum_{j=1}^{p} \frac{1}{j!} D^j f(x)[h]^j.$$

Let us denote the Lipschitz constant for the p-th derivative:

$$\|D^p f(x) - D^p f(y)\| \leq L_p \|x - y\|,$$

such that by Taylor's theorem, we have

$$\begin{aligned} |f(y) - \Phi_{x,p}(y)| &\leq \frac{L_p}{(p+1)!} \|y-x\|^{p+1} \\ |\nabla f(y) - \Phi_{x,p-1}(y)| &\leq \frac{L_p}{p!} \|y-x\|^p \\ \left|\nabla^2 f(y) - \Phi_{x,p-1}(y)\right| &\leq \frac{L_p}{(p-1)!} \|y-x\|^{p-1} \\ &\vdots \end{aligned}$$

Now, how should we expect local search methods that have the p-th order information to perform in general? As we'll see, we still get sub-linear convergence, but as one might expect, the performance scales polynomially with the differentiability of the objective.

3.1 "Worst Functions in the World"

To construct the family of functions that contains the "worst function in the world", let us assume that dom $= \mathbb{R}^n$. Let us define a function

$$g_{p+1}(x) = \frac{1}{p+1} \sum_{i=1}^{n} |x_i|^{p+1}, \quad x \in \mathbb{R}^n$$

such that g_{p+1} is convex and p times differentiable, where

$$D^k g_{p+1}(x)[h]^k = \begin{cases} \frac{p!}{(p+1-k)!} \sum_{i=1}^{n} |x_i|^{p+1-k} h_i^k, & k \text{ even} \\ \frac{p!}{(p+1-k)!} \sum_{i=1}^{n} |x_i|^{p-k} x_i h_i^k, & k \text{ odd} \end{cases} . \tag{3.1}$$

To calculate the Lipschitz constant for $D^p g_{p+1}(x)$, we use the Cauchy-Schwarz inequality: given $x, y, h \in \mathbb{R}^n$

$$\begin{aligned} |D^p g_{p+1}(x)[h]^p - D^p g_{p+1}(y)[h]^p| &\leq p! \, \|x-y\| \left(\sum_{i=1}^{n} h_i^{2p} \right)^{1/2} \\ &\leq p! \, \|x-y\| \left(\left(\sum_{i=1}^{n} h_i^2 \right)^{2p/2} \right)^{1/2} \\ &= p! \, \|x-y\| \, \|h\|^p \end{aligned}$$

Therefore, if $\|h\| \leq 1$, then we get

$$L_p(g_{p+1}) \leq p!. \tag{3.2}$$

Keeping $g_{p+1}(x)$ in mind, we introduce the matrices that will complete the construction of the family of functions

$$U_k = \begin{bmatrix} 1 & -1 & 0 & & & \\ 0 & 1 & -1 & & \vec{0} & \\ 0 & 0 & 1 & & & \\ & & & \ddots & & \\ & \vec{0} & & & 1 & -1 \\ & & & & 0 & 1 \end{bmatrix}_{k \times k} \quad \text{noting} \quad U_k^{-1} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 0 & 1 & 1 & \cdots & 1 \\ 0 & 0 & 1 & & \\ \vdots & \vdots & & \ddots & \\ 0 & 0 & & & 1 \end{bmatrix}_{k \times k}$$

$$A_k := \begin{bmatrix} U_k & 0 \\ 0 & I_{n-k} \end{bmatrix}_{n \times n}.$$

Observe that by construction each A_k is invertible. Furthermore, by simple calculations we find the operator norm of A_k is upper bounded: $\|A_k\|_{op} \leq 2$.

We now propose the family of functions $\mathcal{F} = \{f_k\}$:

$$f_k(x) := g_{p+1}(A_k x) - e_1^\top x. \tag{3.3}$$

Since f_k is the sum of a convex function and a linear function, it is convex. The reason why we had to go the extra mile to define A_k is so that the minimizer of f_k can be explicitly derived from the first-order optimality condition

$$\begin{aligned}\nabla f_k(x) &= A_k^\top \nabla g_{p+1}(A_k x) - e_1 \\ \nabla g_{p+1}(A_k x) &= A_k^{-\top} e_1 \\ &= [\underbrace{1, 1, \ldots, 1}_{k \text{ times}}, 0, 0 \ldots]^\top \\ &:= e_{[k]}.\end{aligned}$$

Using the definition of g_{p+1}, we can solve this equation

$$\nabla g_{p+1}(A_k x_k^*) = \sum_{i=1}^n |(A_k x_k^*)_i|^{p-1} (A_k x_k^*)_i = \begin{cases} 1 & i \in [k] \\ 0 & i > k \end{cases} \tag{3.1}$$

$$\begin{aligned}\implies (A_k x_k^*)_i &= \begin{cases} 1 & i \in [k] \\ 0 & i > k \end{cases} = e_[k] \\ x_k^* &= A_k^{-1} e_{[k]} \\ &= [k, k-1, \ldots, 1, 0, 0, \ldots]^\top. \end{aligned} \tag{3.4}$$

Denoting the optimal solution of f_k: x_k^* and plugging it in, we get

$$f_k^* = g_{p+1}(e_{[k]}) - e_1^\top x = \frac{k}{p+1} - k = \frac{-kp}{p+1} \tag{3.5}$$

$$\|x_k^*\|^2 = \sum_{j=1}^k j^2 = \frac{k(k+1)(2k+1)}{6} \leq \frac{(k+1)^3}{3}. \tag{3.6}$$

For the next part of the construction, we revisit and refine the key Subspace Expansion lemma from Section 1. However, now we must be a little more intricate in formulating our assumptions on the class of tensor methods we want to consider. Since we are trying to characterize local search tensor methods, we want the method to take as input a point, and then return a new point based on up to the p-th order of information at the previous point. In other words, we assume that as a subroutine, given location $\overline{x}$, the method can find the stationary solutions to

$$\phi_{c,\gamma,m}(h) = \sum_{j=1}^p c_j D^j f(\overline{x})[h]^j + \gamma \|h\|^m$$

where $c \in \mathbb{R}^p$, $\gamma > 0$, and $m > p$. Notice that c defines a linear combination of the tensors, whereas γ and m control the contribution of the higher order terms. Clearly the locally optimal direction from $\overline{x}$ can be written as a stationary point of $\phi_{c,\gamma,m}(h)$ for some choice of c, γ, m. Furthermore, $\phi_{c,\gamma,m}(h)$ is a polynomial in terms of h for even m, and still a relatively simple composite function for odd m, so it isn't too optimistic to assume that we can find the

stationary points of $\phi_{c,\gamma,m}$. We now define the set $\Gamma_{\overline{x},f}(c,\gamma,m)$ containing all the stationary points of $\phi_{c,\gamma,m}$ given function f and location $\overline{x}$. We now define the subspace

$$S_f(\overline{x}) = \mathbf{span}\left(\Gamma_{\overline{x},f}(c,\gamma,m) : c \in \mathbb{R}^p, \gamma > 0, m > p\right). \tag{3.7}$$

We are now ready to characterize the class of methods that we care about

Definition 3.1 *The class of methods $\mathcal{M}$ contain all methods that generate iterate points x_{k+1} that satisfy*

$$x_{k+1} = x_0 + \sum_{i=0}^{k} S_f(x_i).$$

Observe that this definition captures most familiar first and second order methods, as well as tensor methods (supposedly; I'm not personally familiar). In particular, the class of methods from Definition 1.3 are contained in $\mathcal{M}$. We now prove the following subspace expansion lemma for the class of "worst functions" $\mathcal{F}$. Let us define $\mathbb{R}^n_{[k]} = \mathbf{span}\left(e_1, \ldots, e_k\right)$.

Lemma 3.2 *Let $x_0 = 0$. Any tensor method from $\mathcal{M}$ minimizing $f_t \in \mathcal{F}$ generates points $\{x_k\}$ that satisfy*

$$x_{k+1} \in \sum_{i=0}^{k} S_{f_t}(x_i) \subseteq \mathbb{R}^n_{[k]}, \quad 0 \le k \le t-1.$$

Proof of Lemma 3.2: The proof hinges on the construction of A_k and its upper triangular structure. We prove the lemma by induction.

- Base case: $x_0 = 0$.

$$\begin{aligned} \nabla f_t(x_0) &= -e_1 \\ D^j f(x_0)[h]^j &= 0, \quad j = 2, \ldots, p. \end{aligned}$$

  Therefore, $x_1 \in \mathbb{R}^n_{[1]}$.

- Induction hypothesis: $x_k \in R^n_{[k]}$. Observe that $A_t x_k \in \mathbb{R}^n_{[k]}$ because A_t is upper triangular. Therefore, we have

$$\begin{aligned} Df_t(x_k)[h] &= Dg_{p+1}(A_t x_k)[A_t h] - h_1 \\ &= \sum_{i=1}^{n} \underbrace{|(A_t x_k)_i|^{p-1} (A_t x_k)_i}_{\text{doesn't depend on } h} (A_t h)_i - h_1 \\ &:= \sum_{i=1}^{k} d_{i,1} e_i^\top (A_t h) - h_1 \quad \text{since } (A_t x_k)_i = 0,\ i > k \\ D^j f_t(x_k)[h]^j &= Dg_{p+1}(A_t x_k)[A_t h] \\ &= \frac{p!}{(p+1-j)!} \sum_{i=1}^{n} |(A_t x_k)_i|^{p+1-j} (A_t h)_i^j, \quad j \ge 2 \\ &:= \sum_{i=1}^{k} d_{i,j} (e_i^\top (A_t h))^j \end{aligned}$$

The stationary points of $\phi_{c,\gamma,m}$ are a linear combination of the gradients of the above directional derivatives. We simply need to verify that the gradient of each lies in $\mathbb{R}^n_{[k+1]}$. We achieve this through routine calculation: observe that $A_t^\top$ is lower bi-diagonal and hence $A_t^\top e_k \in \mathbb{R}^n_{[k+1]}$:

$$\nabla_h D f_t(x_k)[h] = \sum_{i=1}^{k} d_{i,1} A_t^\top e_i - e_1 \in \mathbb{R}^n_{[k+1]}$$

$$\nabla_h D^j f_t(x_k)[h]^j = \sum_{i=1}^{k} j d_{i,j} (e_i^\top (A_t h))^{j-1} (A_t^\top e_i) \in \mathbb{R}^n_{[k+1]}.$$

Observe that the value of $\gamma \|h\|^m$ is radially symmetric around the origin, and so the stationary points of $\phi_{c,\gamma,m}(h)$ given any c, γ, m will always lie in $\mathbb{R}^n_{[k+1]}$. Specifically, any point the tensor method picks for $x_{k+1} \in \mathbb{R}^n_{[k+1]}$, which completes the induction step.

We have proven that a broad class of tensor methods can only expand the search space by one dimension per iteration on $\mathcal{F}$. We can combine this fact with the special structure of the "worst" function family to get the following technical lemma.

Lemma 3.3 *Given $f_a, f_b \in \mathcal{F}$ where $a < b$, then for $x \in \mathbb{R}^n_{[a]}$*

$$f_a(x) = f_b(x) \tag{3.8}$$

and in particular

$$|f_b(x) - f_b^*| = |f_a(x) - f_b^*| \geq |f_a^* - f_b^*| = f_a^* - f_b^*, \tag{3.9}$$

since we know $f_b^* \leq f_a^*$ by combining 3.8 and the fact that $x_a^* \in \mathbb{R}^n_{[a]}$ by our construction.

We are now ready to prove the main lower complexity bound. The proof in Nesterov's paper [8] seems to be flawed, or at least over-complicated. The proof of the following theorem is my own (which runs a real risk of also being flawed).

Theorem 3.4 *For any tensor method in $\mathcal{M}$, there exist functions f that are convex and p times differentiable such that for t where $2t + 1 \leq n$, we have the following lower bound on the rate of convergence*

$$\min_{0 \leq k \leq t} |f(x_k) - f^*| \geq \left|f(x_k) - f_{2t+1}^*\right| \geq \frac{2^{-\frac{3p+3}{2}} \frac{p}{p+1} \left\|x_0 - x_{2t+1}^*\right\|^{p+1}}{(t+1)^{\frac{3p+1}{2}}} = \Omega\left(1/t^{\frac{3p+1}{2}}\right).$$

Notice that we did not need to use any Lipschitz constants here.

Proof of Theorem 3.4: let us set

$$f := f_{2t+1} \in \mathcal{F}.$$

From Lemmas 3.2 and 3.3, we know that for $0 \leq k \leq t$, we have that $x_k \in \mathbb{R}^n_{[k]}$, and thus

$$\begin{aligned}
\left|f_{2t+1}(x_k) - f^*_{2t+1}\right| &= \left|f_k(x_k) - f^*_{2t+1}\right| \\
&\geq f^*_k - f^*_{2t+1} \\
&\geq f^*_t - f^*_{2t+1} \\
&= \frac{-tp}{p+1} + \frac{2tp+p}{p+1} \quad \text{equation 3.5} \\
&= \frac{p(t+1)}{p+1}.
\end{aligned}$$

and from 3.6 we also have that

$$\left\|x^*_{2t+1}\right\|^{p+1} \leq (2t+2)^{\frac{3(p+1)}{2}} = 2^{\frac{3p+3}{2}}(t+1)^{\frac{3p+3}{2}}.$$

We observe that without loss of generality, we can set $x_0 = 0$, because given a tensor method, we can adjust it so that it performs its search offset by x_0. We now have the following series of inequalities:

$$\begin{aligned}
\frac{\left|f(x_k) - f^*_{2t+1}\right|}{\left\|x_0 - x^*_{2t+1}\right\|^{p+1}} &\geq \frac{\frac{p(t+1)}{p+1}}{2^{\frac{3p+3}{2}}(t+1)^{\frac{3p+3}{2}}} \\
&= \frac{2^{\frac{1}{\frac{3p+3}{2}}\frac{p}{p+1}}}{(t+1)^{\frac{3p+1}{2}}} \\
\left|f(x_k) - f^*_{2t+1}\right| &\geq \frac{2^{-\frac{3p+3}{2}\frac{p}{p+1}}\left\|x_0 - x^*_{2t+1}\right\|^{p+1}}{(t+1)^{\frac{3p+1}{2}}} \\
&= \Omega\left(1/t^{\frac{3p+1}{2}}\right).
\end{aligned}$$

This completes the construction of a general family of "worst functions in the world" given local search methods of any order!

References

[1] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences, 2 (1), 183-202, 2009.

[2] J.F. Cai, E.J. Candés, Z. Shen, *A Singular Value Thresholding Algorithm for Matrix Completion*, SIAM J. Optim., 20 (4), 1956-1982, 2010.

[3] E.J. Candés, X. Li, Y. Ma, J. Wright, *Robust Principal Component Analysis?* Journal of the ACM, 58 (3), Article No. 11, May 2011.

[4] D. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory, 52 (4), 1289-1306, 2006.

[5] Yu. Nesterov, *A method for unconstrained convex minimization problem with the rate of convergence* $O(1/k^2)$, Dokl. Akad. Nauk SSSR (translated as Soviet Math. Docl.), 269: 543-547, 1983.

[6] Yu. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Science & Business Media, Vol. 87, 2013.

[7] Yu. Nesterov, *Gradient methods for minimizing composite functions*, Mathematical Programming, 140 (1), 125-161, 2013.

[8] Yu. Nesterov, *Implementable Tensor Methods in Unconstrained Convex Optimization*, CORE Discussion, 2018.

[9] B. Recht, M. Fazel and P.A. Parrilo, *Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization*, SIAM Review, 52 (3), 471-501, 2010.

[10] R.T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14, 877-898, 1976.

[11] R. Tibshirani, *Regression Shrinkage and Selection via the lasso*, Journal of the Royal Statistical Society B, 58 (1) 267-88, 1996.

[12] A.M. Tillmann, M.E. Pfetsch, *The Computational Complexity of the Restricted Isometry Property, the Nullspace Property, and Related Concepts in Compressed Sensing*, IEEE Transactions on Information Theory, 60 (2), 1248-1259, 2013.