BARRIER SMOOTHING FOR NONSMOOTH CONVEX MINIMIZATION

Quoc Tran-Dinh, Yen-Huan Li and Volkan Cevher

LIONS, EPFL, Lausanne, Switzerland

ABSTRACT

This paper proposes a smoothing technique for nonsmooth convex minimization using *self-concordant barriers*. To illustrate the main ideas, we compare our technique and the proximity smoothing approach [1] via the classical gradient method on both the theoretical and numerical aspects. While the barrier smoothing approach maintains the sublinear-convergence rate, it affords a *new analytic step size*, which significantly enhances the practical convergence of the gradient method as compared to proximity smoothing.

Index Terms— Self-concordant barrier, smoothing, gradient method, nonsmooth convex optimization.

1. INTRODUCTION

In this paper, we consider a stylized convex minimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n}\left\{f(\mathbf{x}) := \max_{\mathbf{y}\in\mathcal{Y}}\left\{\langle\mathbf{A}\mathbf{x},\mathbf{y}\rangle - G(\mathbf{y})\right\} + \langle\mathbf{c},\mathbf{x}\rangle\right\}, \qquad (1)$$

where $\mathbf{A}\in\mathbb{R}^{m\times n}$, $\mathcal{Y}$ is a closed convex set in $\mathbb{R}^m$, $\mathbf{c}\in\mathbb{R}^n$ and G is a convex function. In general, problem (1) is nonsmooth, which is of interest in this paper, except when G is strictly convex.

In principle, the problem (1) can be solved by the *smoothing via proximity functions* technique, which has attracted a great deal of attention during the last two decades due to its efficiency in signal processing and machine learning applications [2, 3, 4, 5, 6]. This pioneering work of Nesterov [1] leverages smoothing via proximity functions within a *fast gradient scheme*, which features a theoretically optimal convergence rate. This technique is commonly referred to under the name of "Nesterov's smoothing technique."

In Nesterov's smoothing technique, we assume that $\mathcal{Y}$ is bounded and $p_{\mathcal{Y}}$ is a proximity function of $\mathcal{Y}$, which is nonnegative and strongly convex on $\mathcal{Y}$ with the parameter 1. As a result, we use the following smoothed function in optimization instead:

$$f_\tau(\mathbf{x}) := \max_{\mathbf{y}\in\mathcal{Y}}\left\{\langle\mathbf{A}\mathbf{x},\mathbf{y}\rangle - G(\mathbf{y}) - \tau p_{\mathcal{Y}}(\mathbf{y})\right\} + \langle\mathbf{c},\mathbf{x}\rangle, \qquad (2)$$

where $\tau > 0$ is a smoothness parameter. Nesterov shows in [1] that f_τ is differentiable and its gradient is given by $\nabla f_\tau(\mathbf{x}) = \mathbf{A}^T\mathbf{y}^*_\tau(\mathbf{x}) + \mathbf{c}$, which is Lipschitz continuous with the Lipschitz constant $L_{f_\tau} := \frac{\|\mathbf{A}\|_2^2}{\tau}$, where $\mathbf{y}^*_\tau(\mathbf{x})$ is the unique solution of (2).

Nesterov's smoothing also affords an approximation guarantee:

$$f_\tau(\mathbf{x}) \le f(\mathbf{x}) \le f_\tau(\mathbf{x}) + \tau\mathcal{D}_{\mathcal{Y}}, \text{ where } \mathcal{D}_{\mathcal{Y}} := \max_{\mathbf{y}\in\mathcal{Y}} p_{\mathcal{Y}}(\mathbf{y}). \quad (3)$$

When we solve the smoothed problem (2) with a properly chosen smoothing parameter τ, we can obtain accuracy guarantees on the original problem (1) via (3). To be more concrete, let us apply the classical gradient method to minimize f_τ using the *optimal step size* $\alpha_k := L_{f_\tau}^{-1}$. Starting from $\mathbf{x}^0 \in \mathrm{dom}(f_\tau)$, we generate a sequence $\left\{\mathbf{x}^k\right\}_{k\ge0} \subset \mathrm{dom}(f_\tau)$ as $\mathbf{x}^{k+1} := \mathbf{x}^k - \alpha_k\nabla f_\tau(\mathbf{x}^k)$. It is shown in [7] that the convergence rate of this method is given by

$$f_\tau(\mathbf{x}^k) - f_\tau(\mathbf{x}^*_\tau) \le \frac{2\|\mathbf{A}\|_2^2}{\tau(k+4)}d_0^2, \ k\ge0, \qquad (4)$$

where $d_0 := \left\|\mathbf{x}^0 - \mathbf{x}^*_\tau\right\|_2$ and $\mathbf{x}^*_\tau := \arg\min_{\mathbf{x}} f_\tau(\mathbf{x})$.

However, it is important to note that computing $\nabla f_\tau(\mathbf{x})$ requires solving the convex subproblem in (2) with the constraint $\mathbf{y}\in\mathcal{Y}$ in general. Moreover, the estimate (3) depends on $\mathcal{D}_{\mathcal{Y}}$, which is the prox-diameter of $\mathcal{Y}$. Depending on the choice of $p_{\mathcal{Y}}$, this quantity may be large, which prevents the application of the gradient method.

In this paper, we further assume that $\mathcal{Y}$ is endowed with a self-concordant barrier $b_{\mathcal{Y}}$ defined as follows:

Definition 1.1 (see, e.g., [8, 7])**.** *A convex function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be self-concordant with parameter $M \ge 0$, if $|\varphi'''(t)| \le M\varphi''(t)^{3/2}$, where $\varphi(t) := f(\mathbf{x}+t\mathbf{v})$ for all $t\in\mathbb{R}$, $\mathbf{x}\in\mathrm{dom}(f)$ and $\mathbf{v}\in\mathbb{R}^n$ such that $\mathbf{x}+t\mathbf{v}\in\mathrm{dom}(f)$. If $M = 2$, then f is said to be standard self-concordant. A standard self-concordant function f is a ν-self-concordant barrier for a given convex set Ω with parameter $\nu > 0$, when φ also satisfies $|\varphi'(t)| \le \sqrt{\nu}\varphi''(t)^{1/2}$ and $f(\mathbf{x}) \to +\infty$ as $\mathbf{x}\to\partial\Omega$, the boundary of Ω.*

Several sets $\mathcal{Y}$ are endowed with a self-concordant barrier. For instance, the orthogonal cone $\mathbb{R}^n_+$, the Lorentz cone, the symmetric positive semidefinite cone $\mathcal{S}^n_+$, and polyhedrons [8, 7].

To this end, we propose an alternative smoothing technique to Nesterov's smoothing using self-concordant barriers. Let $b_{\mathcal{Y}}$ be a self-concordant barrier of $\mathcal{Y}$, we define

$$f_\sigma(\mathbf{x}) := \max_{\mathbf{y}\in\mathrm{int}(\mathcal{Y})}\left\{\langle\mathbf{A}\mathbf{x},\mathbf{y}\rangle - G(\mathbf{y}) - \sigma b_{\mathcal{Y}}(\mathbf{y})\right\} + \langle\mathbf{c},\mathbf{x}\rangle, \qquad (5)$$

where $\sigma > 0$ is a smoothness parameter, and $\mathrm{int}(\mathcal{Y})$ denotes the interior of $\mathcal{Y}$. We define $\mathrm{dom}(f_\sigma)$ as the domain of f_σ, and $\mathbf{y}^*_\sigma(\mathbf{x})$ as the unique solution of (5).

We show that f_σ is differentiable and its gradient inherits a "Lipschitz-like" property in Section 2. One of the most important features of this method is that computing $\nabla f_\sigma(\mathbf{x})$ requires solving a system of nonlinear equations of the form:

$$\mathbf{A}\mathbf{x} - \nabla G(\mathbf{y}^*_\sigma(\mathbf{x})) - \sigma\nabla b_{\mathcal{Y}}(\mathbf{y}^*_{\mathcal{Y}}(\mathbf{x})) = 0, \qquad (6)$$

provided that G is differentiable. Solving this system usually demands a lower computational cost than the general convex programming problem in (2). In addition, if G is also self-concordant, then solving (6) can be done efficiently [7].

Our contributions: We propose a smoothing technique using self-concordant barriers for structural nonsmooth convex optimization. We illustrate this technique via a classical gradient method with a

This work was supported in part by the European Commission under Grant MIRG-268398, ERC Future Proof and SNF 200021-132548, SNF 200021-146750 and SNF CRSII2-147633.

new analytic step-size update. We show that this method has the same convergence rate $\mathcal{O}\left(1/(\sigma k)\right)$ as in proximity smoothing methods. However, our method allows us to adaptively update the step-size by exploiting the local information of the smoothed objective function and leads to a better performance in practice than using the worst-case step-size. Moreover, the cost-per-iteration is in general lower than in proximity smoothing methods.

Paper organization: The rest of this paper is organized as follows. In the next section, we propose our optimization framework for solving (1). Section 3 compares our method and the proximity smooth approach both in theory and in numerical experiments.

2. OPTIMIZATION FRAMEWORK

In this section, we present key properties of f_σ and illustrate how we can leverage them within the classical gradient method.

2.1. Smoothing via self-concordant barriers

Let $b_{\mathcal{Y}}$ be a given self-concordant barrier of $\mathcal{Y}$ with the parameter $\nu > 0$. We define

$$\mathbf{y}_c^* := \arg\min_{\mathbf{y}\in\mathrm{int}(\mathcal{Y})} b_{\mathcal{Y}}(\mathbf{y}), \tag{7}$$

the analytic center of the set $\mathcal{Y}$. It is well-known that if $\mathcal{Y}$ is bounded then $\mathbf{y}_c^*$ exists and is unique. Without loss of generality, we assume that $b_{\mathcal{Y}}(\mathbf{y}_c^*) = 0$; otherwise, we can shift $b_{\mathcal{Y}}$ by $\tilde{b}_{\mathcal{Y}}(\mathbf{y}) := b_{\mathcal{Y}}(\mathbf{y}) - b_{\mathcal{Y}}(\mathbf{y}_c^*)$. For given $\mathbf{x} \in \mathrm{dom}(f_\sigma)$, we also define the following quantity:

$$c_A(\mathbf{x}) := \|\mathbf{A}^T\nabla^2 b_{\mathcal{Y}}(\mathbf{y}_\sigma^*(\mathbf{x}))^{-1}\mathbf{A}\|_2^{1/2}, \tag{8}$$

and $f_c(\mathbf{x}) := \langle \mathbf{A}\mathbf{x}, \mathbf{y}_c^*\rangle - G(\mathbf{y}_c^*) + \langle \mathbf{c}, \mathbf{x}\rangle$. We recall the following key properties of $f_\sigma(\cdot)$, whose proof can be obtained as in [9].

Lemma 2.1. *Let f be a function given by* (1) *and f_σ be defined by* (5)*. Then, for any $\sigma > 0$, f_σ is convex and*

$$f_\sigma(\mathbf{x}) \le f(\mathbf{x}) \le f_\sigma(\mathbf{x}) + \sigma\nu\left\{1 + \left[\ln\left(\frac{f(\mathbf{x}) - f_c(\mathbf{x})}{\sigma\nu}\right)\right]_+\right\}, \tag{9}$$

where $[a]_+ := \max\{0, a\}$. Moreover, f_σ is differentiable in $\mathrm{dom}(f_\sigma)$ *and its gradient is given by $\nabla f_\sigma(\mathbf{x}) = \mathbf{A}^T\mathbf{y}_\sigma^*(\mathbf{x}) + \mathbf{c}$, which satisfies, for any x and $\hat{x}$ in* $\mathrm{dom}(f_\sigma)$*,*

$$\langle \nabla f_\sigma(\mathbf{x}) - \nabla f_\sigma(\hat{\mathbf{x}}), \mathbf{x} - \hat{\mathbf{x}}\rangle \ge \frac{\sigma\left\|\nabla f_\sigma(\mathbf{x}) - \nabla f_\sigma(\hat{\mathbf{x}})\right\|_2^2}{c_A\left(c_A + \|\nabla f_\sigma(\mathbf{x}) - \nabla f_\sigma(\hat{\mathbf{x}})\|_2\right)}, \tag{10}$$

where $c_A := c_A(\mathbf{x})$. Consequently, if $c_A\left\|\mathbf{x} - \hat{\mathbf{x}}\right\|_2 < \sigma$ then

$$f_\sigma(\hat{\mathbf{x}}) \le f_\sigma(\mathbf{x}) + \langle \nabla f_\sigma(\mathbf{x}), \hat{\mathbf{x}} - \mathbf{x}\rangle + \sigma\omega_*\left(\sigma^{-1}c_A\left\|\hat{\mathbf{x}} - \mathbf{x}\right\|_2\right), \tag{11}$$

where $\omega_*(\tau) = -\tau - \ln(1 - \tau) \le \frac{\tau^2}{2(1-\tau)}$ for $\tau \in [0, 1)$.

The estimate (9) shows that for any point $\mathbf{x}$ such that $f(\mathbf{x}) - f_c(\mathbf{x}) \le \sigma\nu e^\rho$, $|f_\sigma(\mathbf{x}) - f(\mathbf{x})| \le (1 + \rho)\sigma\nu \to 0^+$ as $\sigma \downarrow 0^+$ for any $\rho > 0$. The second estimate in Lemma 2.1 plays a similar role as the Lipschitz gradient of f_σ, but locally.

Next, we show that $c_A(\mathbf{x})$ is bounded.

Lemma 2.2. *The function $c_A(\cdot)$ defined by* (8) *is bounded on* $\mathrm{dom}(f_\sigma)$*, e.g.* $c_A(\mathbf{x}) \le \bar{c}_A := (\nu+2\sqrt{\nu})\left\|\mathbf{A}^T\nabla^2 b_{\mathcal{Y}}(\mathbf{y}_c^*)^{-1}\mathbf{A}\right\|_2^{1/2}$.

Proof. Apply [7, Corollary 4.2.1]. □

Remark 2.1. *We note that, in several examples, the constant $\bar{c}_A$ defined in Lemma 2.2 can be worse than the actual upper bound of $c_A(\cdot)$. For example, if we consider $f(\mathbf{x}) := \left\|\mathbf{A}\mathbf{x} - \mathbf{b}\right\|_1 =$ $\max_{\|\mathbf{y}\|_\infty\le 1}\langle \mathbf{A}\mathbf{x} - \mathbf{b}, \mathbf{y}\rangle$, then we can choose the barrier function $b_{\mathcal{Y}}(\mathbf{y}) := -\sum_{i=1}^m \ln(1 - \mathbf{y}_i^2)$. In this case, it is easy to see that $\mathbf{y}_c^* = 0^T$. Consequently, $\nabla^2 b_{\mathcal{Y}}(\mathbf{y}_c^*) \succeq 2\mathbb{I}$, which leads to* $c_A(\mathbf{x}) \le \frac{1}{\sqrt{2}}\left\|\mathbf{A}\right\|_2$.

Finally, we consider the smoothed problem of (1) and its optimality condition:

$$f_\sigma^* \equiv f_\sigma(\mathbf{x}_\sigma^*) := \min_{\mathbf{x}\in\mathbb{R}^n} f_\sigma(\mathbf{x}) \Leftrightarrow \nabla f_\sigma(\mathbf{x}_\sigma^*) = 0. \tag{12}$$

Here, we denote by $\mathbf{x}_\sigma^*$ the unique solution of (12). By Lemma 2.1, we can see that, within an accuracy level $\sigma > 0$, $\mathbf{x}_\sigma^*$ approximates the solution $\mathbf{x}^*$ of (1).

2.2. The gradient method with analytic step-size

Let us apply the gradient method to solve (12). By exploiting the properties of f_σ in Lemma 2.1, we can derive a new analytic step-size for this gradient scheme.

Let $\mathbf{x}^0 \in \mathrm{dom}(f_\sigma)$, the gradient scheme for solving (12) is defined as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k\nabla f_\sigma(\mathbf{x}^k),\ k \ge 0. \tag{13}$$

where the step size $\alpha_k \in (0, 1]$ will be defined later. Let $\mathbf{d}^k := -\nabla f_\sigma(\mathbf{x}^k)$ be the antigradient direction and $r_k := \|\mathbf{d}^k\|_2$. As shown in Lemma 2.1 that the gradient $\nabla f_\sigma(\mathbf{x}^k)$ is given by $\nabla f_\sigma(\mathbf{x}^k) = \mathbf{A}^T\mathbf{y}_\sigma^*(\mathbf{x}^k) + \mathbf{c}$, where $\mathbf{y}_\sigma^*(\mathbf{x}^k)$ is obtained from the optimality condition (6). The following lemma shows how to derive the step size α_k in (13).

Lemma 2.3. *Let $\left\{\mathbf{x}^k\right\}_{k\ge 0}$ be the sequence generated by* (13)*. If α_k is chosen as $\alpha_k := \sigma/(c_A^k(c_A^k + r_k))$, then $\left\{\mathbf{x}^k\right\} \subset \mathrm{dom}(f_\sigma)$ and*

$$f_\sigma(\mathbf{x}^{k+1}) \le f_\sigma(\mathbf{x}^k) - \sigma\omega\left(r_k/c_A^k\right), \tag{14}$$

where $\omega(\tau) := \tau - \ln(1 + \tau) > 0$ for $\tau > 0$ and $c_A^k := c_A(\mathbf{x}^k)$. Moreover, the step size α_k is optimal.

Proof. We obtain by (11) that $f_\sigma(\mathbf{x}^{k+1}) \le f_\sigma(\mathbf{x}^k) - \varphi(\alpha_k)$, where $\varphi(\alpha) := r_k^2\alpha - \sigma\omega_*\left(\sigma^{-1}c_A^k r_k\alpha\right)$. By maximizing φ over $[0, 1]$, we obtain the optimal step size $\alpha_k := \sigma/(c_A^k(c_A^k + r_k))$, which satisfies $\alpha_k < \sigma/(c_A^k r_k)$. The last condition shows that $\mathbf{x}^{k+1} \in \mathrm{dom}(f_\sigma)$. Moreover, we have $\varphi(\alpha_k) = \sigma\omega(r_k/c_A^k)$. □

Based on the step-size α_k in Lemma 2.3, we can describe a gradient method for solving (1) as follows.

Algorithm 1 (*Barrier-gradient method*)

```
Inputs: Fix $\sigma > 0$ and a tolerance $\varepsilon > 0$. Take $\mathbf{x}^0 \in \mathrm{dom}(f_\sigma)$.
for $k = 0$ to $k_{\max}$ do
    1. Compute $\mathbf{y}_\sigma^*(\mathbf{x}^k)$ by solving (6). Then, compute $\nabla f_\sigma(\mathbf{x}^k) := \mathbf{A}^T\mathbf{y}_\sigma^*(\mathbf{x}^k) + \mathbf{c}$.
    2. Compute $r_k := \|\nabla f_\sigma(\mathbf{x}^k)\|_2$ and $c_A^k := c_A(\mathbf{x}^k)$ as in (8).
    3. If $r_k \le \varepsilon$, then terminate.
    4. Otherwise, update $\mathbf{x}^{k+1} := \mathbf{x}^k - \alpha_k\nabla f_\sigma(\mathbf{x}^k)$, where $\alpha_k := \sigma/(c_A^k(c_A^k + r_k))$.
end for
```

We note that, at each iteration of Algorithm 1 we have to compute c_A^k, which requires $\nabla^2 b_{\mathcal{Y}}(\mathbf{y}_\sigma^*(\mathbf{x}^k))^{-1}$ and matrix multiplications. This quantity can be computed in $\mathcal{O}(n^2)$ operations by the power method. Instead of using c_A^k, we can use its upper bound $\bar{c}_A$ as given in Lemma 2.2. In this case, the step size α_k can be replaced by $\bar{\alpha}_k := \sigma/(\bar{c}_A(\bar{c}_A + r_k)$ without any additional computation. This becomes the worst-case step-size.

Now, we prove the convergence and the convergence rate of Algorithm 1.

Theorem 2.1. *Let* $\left\{\mathbf{x}^k\right\}_{k\geq 0}$ *be the sequence generated by Algorithm 1. Then, the number of iterations needed to reach the point* $\mathbf{x}^0 \in \mathrm{dom}(f_\sigma)$ *such that* $f_\sigma(\mathbf{x}^0) - f_\sigma^* \leq \bar{c}_A \left\|\mathbf{x}^0 - \mathbf{x}_\sigma^*\right\|$ *does not exceed* $\left\lfloor \frac{f_\sigma(\mathbf{x}^0) - f_\sigma^*}{\sigma\omega(1)} \right\rfloor + 1$. *If* $\mathbf{x}^0$ *is chosen such that* $f_\sigma(\mathbf{x}^0) - f_\sigma^* \leq \bar{c}_A \left\|\mathbf{x}^0 - \mathbf{x}_\sigma^*\right\|$ *then*

$$f_\sigma(\mathbf{x}^k) - f_\sigma^* \leq \frac{4\bar{c}_A^2 \left\|\mathbf{x}^0 - \mathbf{x}_\sigma^*\right\|^2}{\sigma k}, \ \forall k \geq 1. \tag{15}$$

Proof. Let $e_k = \|\mathbf{x}^k - \mathbf{x}_\sigma^*\|$. By (13) and $\nabla f_\sigma(\mathbf{x}_\sigma^*) = 0$, we have

$$\begin{aligned} e_{k+1}^2 &= \|x^k + \alpha_k \mathbf{d}^k - \mathbf{x}_\sigma^*\|^2 \\ &= e_k^2 - 2\alpha_k \left\langle \nabla f_\sigma(x^k) - \nabla f_\sigma(\mathbf{x}_\sigma^*), x^k - \mathbf{x}_\sigma^* \right\rangle + \alpha_k^2 r_k^2 \\ &\overset{(10)}{\leq} e_k^2 - 2\alpha_k \frac{\sigma r_k^2}{c_A^k(c_A^k + r_k)} + \alpha_k^2 r_k^2 \\ &\overset{\text{Lemma 2.3}}{\leq} e_k^2 - \alpha_k^2 r_k^2. \end{aligned}$$

Therefore, the sequence $\{e_k\}_{k\geq 0}$ is nonincreasing, i.e., $e_k \leq e_0 = \|\mathbf{x}^0 - \mathbf{x}_\sigma^*\|_2$ for all $k \geq 0$. Let $\Delta_k := f_\sigma(\mathbf{x}^k) - f_\sigma^*$. By the convexity of f_σ and the Cauchy-Schwarz inequality, we can show that $\Delta_k \leq \left\langle \nabla f_\sigma(\mathbf{x}^k), \mathbf{x}^k - \mathbf{x}_\sigma^* \right\rangle \leq e_0 r_k$. On the other hand, from Lemma 2.3 and $c_A^k \leq \bar{c}_A$ we have

$$\Delta_{k+1} \leq \Delta_k - \sigma\omega(r_k/\bar{c}_A) \leq \Delta_k - \sigma\omega\left(\Delta_k/(e_0\bar{c}_A)\right). \tag{16}$$

Since $\omega(\tau) \geq \tau^2/4$ if $\tau \in [0, 1]$. We consider two cases:
Case 1: If $\Delta_k \geq e_0\bar{c}_A$, then (16) implies $\Delta_k \leq \Delta_0 - k\sigma\omega(1)$. Therefore, $k \leq \frac{f_\sigma(\mathbf{x}^0) - f_\sigma(\mathbf{x}^k)}{\sigma\omega(1)} \leq \frac{f_\sigma(\mathbf{x}^0) - f_\sigma^*}{\sigma\omega(1)} \approx \frac{f_\sigma(\mathbf{x}^0) - f_\sigma^*}{0.30685\sigma}$.
Case 2: If $\Delta_k \leq e_0\bar{c}_A$, then we have $\Delta_{k+1} \leq \Delta_k - \sigma\frac{\Delta_k^2}{4(e_0\bar{c}_A)^2}$, which leads to $\Delta_{k+1}^{-1} \geq \Delta_k^{-1} + \frac{\sigma}{4(e_0\bar{c}_A)^2}\frac{\Delta_k}{\Delta_{k+1}} \geq \Delta_k^{-1} + \frac{\sigma}{4(e_0\bar{c}_A)^2} \geq \Delta_0^{-1} + (k+1)\frac{\sigma}{4(e_0\bar{c}_A)^2}$. Therefore, we can show that $\Delta_k \leq \frac{4\bar{c}_A^2 e_0^2 \Delta_0}{4\bar{c}_A^2 e_0^2 + k\Delta_0}$, which implies (15). □

If we choose $\sigma := \frac{\bar{c}_A}{\sqrt{k}}$, then Theorem 2.1 shows that the convergence rate of Algorithm 1 is $\mathcal{O}\left(\frac{\bar{c}_A d_0^2}{\sqrt{k}}\right)$, where $d_0 := \|\mathbf{x}^0 - \mathbf{x}_\sigma^*\|_2$.

3. BARRIER VS. PROXIMITY SMOOTHING

In this section, we compare two smoothing techniques (via proximity functions [1] and via self-concordant barriers) on the gradient method for solving (1).

3.1. Theoretical comparison

Let $\mathbf{H}$ be a lower bound of $\nabla^2 b_{\mathcal{Y}}(\mathbf{y})$, i.e. $\nabla^2 b_{\mathcal{Y}}(\mathbf{y}) \succeq \mathbf{H} \succeq \underline{\mathbf{H}} := (\nu + 2\sqrt{\nu})^{-1}\nabla^2 b_{\mathcal{Y}}(\mathbf{y}_c^*)$ for $\mathbf{y} \in \mathrm{dom}(b_{\mathcal{Y}})$. As mentioned in Remark 2.1, $\mathbf{H}$ is not necessarily identical to $\underline{\mathbf{H}}$. Then the convergence rate of Algorithm 1 is $\mathcal{O}\left(\frac{4\left\|\mathbf{A}^T\mathbf{H}^{-1}\mathbf{A}\right\|_2}{\sigma_k} d_0^2\right)$, provided that $f_\sigma(\mathbf{x}^0) - f_\sigma^* \leq \left\|\mathbf{A}^T\mathbf{H}^{-1}\mathbf{A}\right\|_2 d_0$. While, we have shown that the convergence rate of the gradient method applying to Nesterov's smoother is $\mathcal{O}\left(\frac{2\|\mathbf{A}\|_2^2 d_0^2}{\tau k}\right)$. The overall computational cost is shown in Table 1. We see that the convergence rates in both methods is of the same order with different constants $2\|\mathbf{A}\|_2^2$ and $4\left\|\mathbf{A}^T\mathbf{H}^{-1}\mathbf{A}\right\|_2$, respectively. However, evaluating the gradient ∇f_σ of f_σ requires to solve a system of nonlinear equation, while evaluating ∇f_τ in general needs to solve a general convex program. Since solving a nonlinear system can be done efficiently by Newton-methods [7] combining with a warm-start strategy, the cost-per-iteration in the barrier smoothing method is lower than in the proximity smoothing one in general.

Table 1: Compare two different smoothing techniques

	Barrier smoothing	Proximity smoothing
Convergence	$\mathcal{O}\left(\frac{4\left\|A^T\mathbf{H}^{-1}A\right\|_2 d_0^2}{\sigma k}\right)$	$\mathcal{O}\left(\frac{2\|\mathbf{A}\|_2^2 d_0^2}{\tau k}\right)$
Complexity-per-iteration	Solving a system of nonlinear equations	Solving a general convex program

3.2. Numerical comparison

Now, we compare Algorithm 1 and the standard gradient method with proximity smoother and the optimal constant step-size (proximity smoothing method) in the following two numerical examples.
a) Quadratically constrained quadratic programming (QCQP): The following problem obtained from the minimax formulation of a QCQP problem:

$$f^* := \min_{\mathbf{x}\in\mathbf{R}^n}\left\{f(\mathbf{x}) := \max_{\langle\mathbf{B}\mathbf{y},\mathbf{y}\rangle\leq 1}\left\{\langle\mathbf{A}\mathbf{x} - \mathbf{b}, \mathbf{y}\rangle - \frac{1}{2}\langle\mathbf{Q}\mathbf{y}, \mathbf{y}\rangle\right\} + \langle\mathbf{c}, \mathbf{x}\rangle\right\}, \tag{17}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{b} \in \mathbf{R}^m$, $\mathbf{c} \in \mathbf{R}^n$, $\mathbf{Q}$ is an $m \times m$ symmetric positive semidefinite matrix and $\mathbf{B}$ is an $m \times m$ symmetric positive definite matrix. It is easy to see that $b_{\mathcal{Y}}(\mathbf{y}) = -\log(1 - \mathbf{y}^T\mathbf{B}\mathbf{y}_2^2)$ is the barrier function of the set $\mathcal{Y} := \left\{\mathbf{y} \in \mathbf{R}^m \mid \mathbf{y}^T\mathbf{B}\mathbf{y} \leq 1\right\}$ with $\nu = 2$. After few simple calculations, we can estimate $c_A(\mathbf{x}) \leq \bar{c}_A = (1 + \sqrt{2})\left\|\mathbf{A}^T\mathbf{B}^{-1}\mathbf{A}\right\|_2^{1/2}$.

For the proximity smoothing method, if we choose $p_{\mathcal{Y}}^1(\mathbf{y}) := \frac{1}{2}\|\mathbf{y}\|^2$, then $D_{\mathcal{Y}} := 0.5\lambda_{\max}(\mathbf{B}^{-1})$, which can be very large, and $L_{f_\tau} := \|\mathbf{A}\|^2/\tau$. However, if we choose $p_{\mathcal{Y}}^2(\mathbf{y}) := \frac{1}{2}\mathbf{y}^T\mathbf{B}\mathbf{y}$ then $D_{\mathcal{Y}} := 0.5$ and $L_{f_\tau} := \left\|\mathbf{A}^T\mathbf{B}^{-1}\mathbf{A}\right\|^2/\tau$.

We test both methods on some synthetic data, where all the matrices and vectors are generated randomly using the Gaussian distribution $\mathcal{N}(0, 1)$. Matrix $\mathbf{A}$ is normalized such that $\|\mathbf{A}\|_2 = 1$, $\mathbf{Q}$ is rank-deficient with $\mathrm{rank}(\mathbf{Q}) = \lfloor 0.1m \rfloor$. Matrix $\mathbf{B}$ is positive definite and vector $\mathbf{c}$ is generated as $\mathbf{c} = -\mathbf{A}^T\mathbf{y}_0$, where $\mathbf{y}_0$ is the normalized eigenvector of $\mathbf{B}$ corresponding to the largest eigenvalue. The problem size is $n = \lfloor 0.3m \rfloor$ and $\sigma = \tau = 10^{-2}$.

We run Algorithm 1 and the proximity smoothing method for the case $p_{\mathcal{Y}}^1$ on 3 problem instances. The results are reported in Table 2. As we can see from this table, Algorithm 1 reaches the final solution with high accuracy, while the proximity smoothing method runs up

Table 2: The results of 3 problems after maximum 500 iterations

Method		Barrier smoothing			Proximity smoothing		
m	f^*	it	$f(\mathbf{x}^k)$	$\left\|\mathbf{y}_k^* - \mathbf{y}^*\right\|_2$	it	$f(\mathbf{x}^k)$	$\left\|\mathbf{y}_k^* - \mathbf{y}^*\right\|_2$
100	42.98367	168	42.98367	0.00142	500	44.18773	1.38502
250	53.09777	234	53.09777	0.00114	500	59.31035	2.71808
500	87.16702	384	87.16702	0.00080	500	104.27851	3.42870

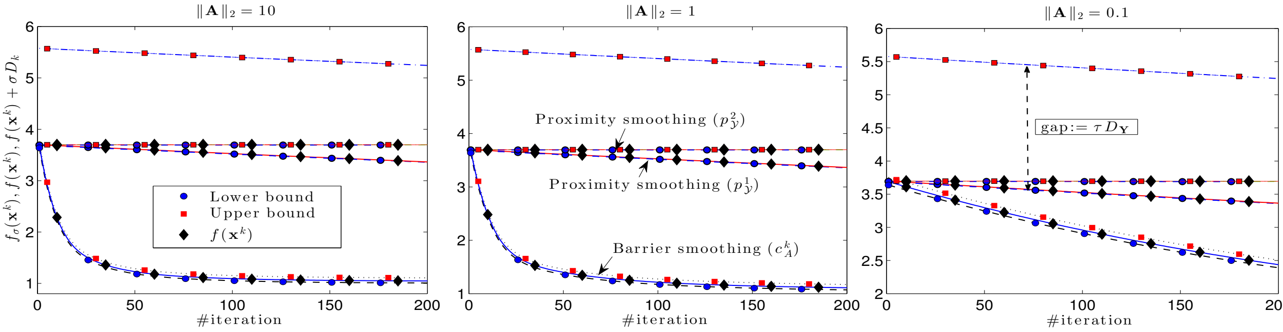

Fig. 1: The lower bound $f_\sigma(\mathbf{x}^k)$, the upper bound $f_\sigma(\mathbf{x}^k) + \sigma D_k$ and the real value $f(\mathbf{x}^k)$ (using $c^k_A, p^1_{\mathcal{Y}}$ and $p^2_{\mathcal{Y}}$, respectively).

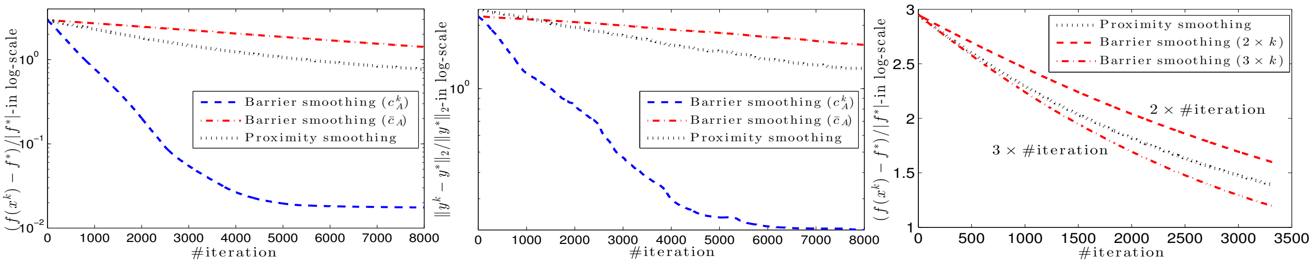

Fig. 2: The convergence of two methods after 8000 iterations ($m = 1000$).

to the maximum number of iterations and still produces a less accurate solution. This happens since Algorithm 1 uses the adaptive step size that captures better the local structure of (17), while the proximity smoothing method runs in the worst-case performance and does not scale well with the change of data.

The convergence behavior and the bound threshold of two smoothing methods on one problem of size $n = 100$ are plotted in Figure 1 for three cases: $\|\mathbf{A}\|_2 = 10$, $\|\mathbf{A}\|_2 = 1$ and $\|\mathbf{A}\|_2 = 0.1$. Here, we show the actual objective value $f(\mathbf{x}^k)$, the lower bound estimate $f_\sigma(\mathbf{x}^k)$ (resp., $f_\tau(\mathbf{x}^k)$ and the upper bound estimate $f_\sigma(\mathbf{x}^k)+\sigma D^1_k$ (reps., $f_\tau(\mathbf{x}^k)+\tau D^2_k$) for three cases, where $D^1_k := \nu + \nu\left[\ln((f(\mathbf{x}^k) - f_c(\mathbf{x}^k))/(\sigma\nu))\right]_+$ (resp., $D^2_k := D_{\mathcal{Y}}$). The lower and upper bound in the proximity smoothing method with respect to $p^2_{\mathcal{Y}}$ is well approximated $f(\mathbf{x}^k)$. However, its performance is also worse than Algorithm 1 in this particular example. It is clear that when $\|\mathbf{A}\|$ is small, the step-size of the proximity smoothing method becomes large and it can accelerate the convergence.

b) Basis pursuit (BP) problem in signal processing. We consider the following constrained BP problem:

$$\max_{\mathbf{y}\in\mathbf{R}^m} \left\{-\|\mathbf{y}\|_1 \mid \mathbf{A}\mathbf{y} - \mathbf{b} = 0,\ \mathbf{y} \in \mathcal{Y} := [\mathbf{l}, \mathbf{u}]\right\}, \tag{18}$$

where $\mathbf{A} \in \mathbb{R}^{n\times m}$, and $0 \in \mathcal{Y}$. The minmax formulation of this problem can be written as

$$\min_{\mathbf{x}\in\mathbf{R}^n} \left\{ f(\mathbf{x}) := \max_{\mathbf{y}\in\mathcal{Y}} \left\{ \left\langle \mathbf{A}^T\mathbf{x}, \mathbf{y} \right\rangle - \|\mathbf{y}\|_1 \right\} - \langle \mathbf{b}, \mathbf{x}\rangle \right\}, \tag{19}$$

The barrier function of $\mathcal{Y}$ is $b_{\mathcal{Y}}(\mathbf{y}) := -\sum_{i=1}^m [\log(\mathbf{y}_i - \mathbf{l}_i) + \log(\mathbf{u}_i - \mathbf{y}_i)]$ with $\nu = 2m$. For Nesterov's smoothing method, we use $p_{\mathcal{Y}}(\mathbf{y}) = \frac{1}{2}\|\mathbf{y}\|_2^2$. With this choice, the gradients ∇f_σ and ∇f_τ in both smoothing methods can be computed in a closed form. Hence, the cost for evaluating these gradient vectors is the same.

We test this algorithm with some synthetic data generated by a random Gaussian process. We choose $\mathcal{Y} := [-\frac{1}{2}, \frac{1}{2}]^m$ and $\mathbf{b} := \mathbf{A}\mathbf{x}_s$, where $\mathbf{x}_s$ is a k-sparse Gaussian random vector, and $\mathbf{A}$ is a Gaussian matrix in $\mathcal{N}(0,1)$ normalized by $1/\sqrt{m}$.

The convergence of Algorithm 1 and the gradient method by using Nesterov's smoother is plotted in Figure 2 for the case $m = 1000$, $k = 0.05n$, $n = 3k$ and $\tau = \sigma = 10^{-2}$. For Algorithm 1 we plot two cases: using adaptive value c^k_A to update the step-size α_k and using $\bar{c}_A$ for updating α_k. The first and the second plot show the relative error of the objective values $f(\mathbf{x}^k)$ and the primal approximation solution $\mathbf{y}^k$ of (18). As we can see, the adaptive step size with c^k_A works much better than the constant step size in the proximity smoothing method. However, this method requires an additional computation for c^k_A with $O(m^2)$ computational effort. The last figure shows that the number of iterations in Nesterov's smoothing method lies in 2 to 3 times the number of iterations in the barrier smoothing method with the worst case step-size using $\bar{c}_A$. This means that the iteration counter k in the right plot of Figure 2 corresponds to $k = l$ in the proximity smoothing method, and $k = 2l$ and $k = 3l$ for the lower and upper curves in the barrier smoothing method, where l is the real iteration counter. We note that the diameters D^1_k and D^2_k in both methods (see Example 1) depend on the number of variables m.

4. CONCLUSIONS

We propose a new smoothing approach for constrained minimax problems of the form (1) using barrier functions. The new smoothing approach has three key advantages: 1) we can efficiently obtain the gradient of the smoothed function via a system of nonlinear equations, 2) we can exploit the local structure of the problem rather than using the global information via an adaptive step-size selection procedure, and 3) we can preserve a dimension independent optimization diameter. As a result, while the analytical complexity of the gradient algorithm based on barrier smoothing is similar to the one using the Nesterov's Lipschitz smoothing approach, the overall arithmetical complexity is reduced. Our future work is to extend this theory to the accelerating method which maintains the same convergence rate $\mathcal{O}(1/k)$ as in proximity smoothing method [1], where k is the iteration counter.

5. REFERENCES

[1] Y. Nesterov, "Smooth minimization of non-smooth functions," *Math. Program.*, vol. 103, no. 1, pp. 127–152, 2005.

[2] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[3] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imaging Vis.*, vol. 40, no. 1, pp. 120–145, 2011.

[4] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Model. Simul.*, vol. 4, pp. 1168–1200, 2005.

[5] Y. Nesterov, "Gradient methods for minimizing composite objective function," CORE Discussion Paper 2007/76, CORE/UCL, 2007.

[6] Q. Tran-Dinh, A. Kyrillidis, and V. Cevher, "Composite self-concordant minimization," Tech. Rep., LIONS, EPFL, 2013.

[7] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*, vol. 87 of *Applied Optimization*, Kluwer Academic Publishers, 2004.

[8] Y. Nesterov and A. Nemirovski, *Interior-point Polynomial Algorithms in Convex Programming*, Society for Industrial Mathematics, 1994.

[9] Q. Tran-Dinh, *Sequential Convex Programming and Decomposition Approaches for Nonlinear Optimization*, Phd thesis, Arenberg Doctoral School, KU Leuven, Nov. 2012.