Analysis of the Frank-Wolfe Method for Convex Composite Optimization involving a Logarithmically-Homogeneous Barrier

Renbo Zhao

MIT Operations Research Center

Joint work with Robert M. Freund (MIT Sloan School of Management)

SIAM Conference on Optimization
July, 2021

Problem Statement

Consider the following convex composite optimization problem:

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \qquad \texttt{(P)}$$

Problem Statement

Consider the following convex composite optimization problem:

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \tag{P}$$

- $f : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a θ-logarithmically-homogeneous self-concordant barrier (θ-LHSCB) for some regular cone $\mathcal{K} \subseteq \mathbb{R}^m$,

Problem Statement

Consider the following convex composite optimization problem:

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \tag{P}$$

- $f : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a θ-logarithmically-homogeneous self-concordant barrier (θ-LHSCB) for some regular cone $\mathcal{K} \subseteq \mathbb{R}^m$,
- $\mathsf{A} : \mathbb{R}^n \to \mathbb{R}^m$ is a linear operator (not necessarily invertible),

Problem Statement

Consider the following convex composite optimization problem:

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \tag{P}$$

- $f : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a θ-logarithmically-homogeneous self-concordant barrier (θ-LHSCB) for some regular cone $\mathcal{K} \subseteq \mathbb{R}^m$,
- $\mathsf{A} : \mathbb{R}^n \to \mathbb{R}^m$ is a linear operator (not necessarily invertible),
- $h : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a proper, closed and convex (but possibly non-smooth) function, and $\mathsf{dom}\, h$ is nonempty convex and compact.

Problem Statement

Consider the following convex composite optimization problem:

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \tag{P}$$

- $f : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a θ-logarithmically-homogeneous self-concordant barrier (θ-LHSCB) for some regular cone $\mathcal{K} \subseteq \mathbb{R}^m$,
- $\mathsf{A} : \mathbb{R}^n \to \mathbb{R}^m$ is a linear operator (not necessarily invertible),
- $h : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a proper, closed and convex (but possibly non-smooth) function, and $\mathsf{dom}\, h$ is nonempty convex and compact.
- We recover the traditional problem setting for Frank-Wolfe when h is the indicator function $h := \iota_{\mathcal{X}}$ of a compact convex set $\mathcal{X}$.

Problem Statement

Consider the following convex composite optimization problem:

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \tag{P}$$

- $f : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a θ-logarithmically-homogeneous self-concordant barrier (θ-LHSCB) for some regular cone $\mathcal{K} \subseteq \mathbb{R}^m$,
- $\mathsf{A} : \mathbb{R}^n \to \mathbb{R}^m$ is a linear operator (not necessarily invertible),
- $h : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a proper, closed and convex (but possibly non-smooth) function, and $\mathsf{dom}\, h$ is nonempty convex and compact.
- We recover the traditional problem setting for Frank-Wolfe when h is the indicator function $h := \iota_{\mathcal{X}}$ of a compact convex set $\mathcal{X}$.
- Assume $\mathsf{dom}\, F \neq \emptyset$, so at least one minimizer $x^* \in \mathsf{dom}\, F$ exists, and define $F^* := F(x^*)$.

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x \in \mathcal{X}} f(x) \qquad \texttt{(tP)}$$

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x\in\mathcal{X}} f(x) \qquad \texttt{(tP)}$$

- $\mathcal{X}$ is a nonempty convex and compact set.

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x\in\mathcal{X}} f(x) \qquad \texttt{(tP)}$$

- $\mathcal{X}$ is a nonempty convex and compact set.
- f is L-smooth w.r.t. $\|\cdot\|$ on $\mathcal{X}$, which then implies

$$f(x') \leq f(x) + \langle \nabla f(x), x' - x\rangle + (L/2)\|x' - x\|^2, \quad \forall\, x', x \in \mathcal{X}. \qquad \texttt{(LSm)}$$

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x\in\mathcal{X}} f(x) \qquad \texttt{(tP)}$$

- $\mathcal{X}$ is a nonempty convex and compact set.
- f is L-smooth w.r.t. $\|\cdot\|$ on $\mathcal{X}$, which then implies

$$f(x') \le f(x) + \langle \nabla f(x), x' - x\rangle + (L/2)\|x' - x\|^2, \quad \forall\, x', x \in \mathcal{X}. \qquad \texttt{(LSm)}$$

- At iteration k of FW, $x^k \in \mathcal{X}$ and the method does the following:

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x \in \mathcal{X}} f(x) \tag{tP}$$

- $\mathcal{X}$ is a nonempty convex and compact set.
- f is L-smooth w.r.t. $\|\cdot\|$ on $\mathcal{X}$, which then implies

$$f(x') \le f(x) + \langle \nabla f(x), x' - x \rangle + (L/2)\|x' - x\|^2, \quad \forall\, x', x \in \mathcal{X}. \tag{LSm}$$

- At iteration k of FW, $x^k \in \mathcal{X}$ and the method does the following:
  - Compute

$$v^k \in \arg\min_{x \in \mathcal{X}} \langle \nabla f(x^k), x \rangle$$

by solving a linear-optimization sub-problem.

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x\in\mathcal{X}} f(x) \tag{tP}$$

- $\mathcal{X}$ is a nonempty convex and compact set.
- f is L-smooth w.r.t. $\|\cdot\|$ on $\mathcal{X}$, which then implies

$$f(x') \le f(x) + \langle \nabla f(x), x' - x\rangle + (L/2)\|x' - x\|^2, \quad \forall\, x', x \in \mathcal{X}. \tag{LSm}$$

- At iteration k of FW, $x^k \in \mathcal{X}$ and the method does the following:
  - Compute

    $$v^k \in \arg\min_{x\in\mathcal{X}} \langle \nabla f(x^k), x\rangle$$

    by solving a linear-optimization sub-problem.
  - Determine step-length $\alpha^k \in [0, 1]$.

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x\in\mathcal{X}} f(x) \qquad \texttt{(tP)}$$

- $\mathcal{X}$ is a nonempty convex and compact set.
- f is L-smooth w.r.t. $\|\cdot\|$ on $\mathcal{X}$, which then implies

$$f(x') \le f(x) + \langle \nabla f(x), x'-x\rangle + (L/2)\|x'-x\|^2, \quad \forall\, x', x \in \mathcal{X}. \qquad \texttt{(LSm)}$$

- At iteration k of FW, $x^k \in \mathcal{X}$ and the method does the following:
  - Compute

    $$v^k \in \arg\min_{x\in\mathcal{X}} \langle \nabla f(x^k), x\rangle$$

    by solving a linear-optimization sub-problem.
  - Determine step-length $\alpha^k \in [0,1]$.
  - Update $x^{k+1} = (1-\alpha_k)x^k + \alpha^k v^k$.

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x \in \mathcal{X}} f(x) \qquad \texttt{(tP)}$$

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x \in \mathcal{X}} f(x) \qquad \texttt{(tP)}$$

- The step-size α_k is typically chosen in one of two ways:

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x \in \mathcal{X}} f(x) \qquad \texttt{(tP)}$$

▷ The step-size α_k is typically chosen in one of two ways:

- Fixed step-size, such as the standard step-size $\alpha_k = 2/(k+2)$, or

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x \in \mathcal{X}} f(x) \tag{tP}$$

▷ The step-size α_k is typically chosen in one of two ways:

- Fixed step-size, such as the standard step-size $\alpha_k = 2/(k+2)$, or
- Adaptive step-size, such as $\alpha_k = \min\{G_k/C_k, 1\}$, where

$$G_k := \langle \nabla f(x^k), x^k - v^k \rangle \quad \text{and} \quad C_k := L\|v^k - x^k\|^2.$$

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x\in\mathcal{X}} f(x) \qquad \texttt{(tP)}$$

- ▷ The step-size α_k is typically chosen in one of two ways:
  - Fixed step-size, such as the standard step-size $\alpha_k = 2/(k+2)$, or
  - Adaptive step-size, such as $\alpha_k = \min\{G_k/C_k, 1\}$, where

$$G_k := \langle \nabla f(x^k), x^k - v^k \rangle \quad \text{and} \quad C_k := L\|v^k - x^k\|^2.$$

- ▷ FW is very useful in "sparse" or otherwise "structured" optimization where $\mathcal{X}$ has special structure, e.g., probability simplex or spectrahedron.

Review of the (traditional) Frank-Wolfe (FW) Method

$$\min_{x\in\mathcal{X}} f(x) \tag{tP}$$

▷ The step-size α_k is typically chosen in one of two ways:

- Fixed step-size, such as the standard step-size $\alpha_k = 2/(k+2)$, or
- Adaptive step-size, such as $\alpha_k = \min\{G_k/C_k, 1\}$, where

$$G_k := \langle \nabla f(x^k), x^k - v^k \rangle \quad \text{and} \quad C_k := L\|v^k - x^k\|^2.$$

▷ FW is very useful in "sparse" or otherwise "structured" optimization where $\mathcal{X}$ has special structure, e.g., probability simplex or spectrahedron.

▷ FW has been generalized to the composite setting:

$$\min_{x\in\mathbb{R}^n} [F(x) := f(\mathsf{A}x) + h(x)] \tag{P}$$

in e.g., Bach (2015) and Nesterov (2018), where the subproblem becomes:

$$v^k \in \arg\min_{x\in\mathbb{R}^n} \langle \nabla f(\mathsf{A}x^k), \mathsf{A}x \rangle + h(x).$$

However, note that all of these works assume that f is L-smooth.

Two Motivating Papers

Two Motivating Papers

- Khachiyan, L.G.: Rounding of polytopes in the real number model of computation. *Mathematics of Operations Research* **21**(2), 307–320 (1996) (Elegant analysis of the FW method with exact line-search for D-optimal design)

Two Motivating Papers

- Khachiyan, L.G.: Rounding of polytopes in the real number model of computation. *Mathematics of Operations Research* **21**(2), 307–320 (1996) (Elegant analysis of the FW method with exact line-search for D-optimal design)

- Dvurechensky, P., Ostroukhov, P., Safin, K., Shtern, S., Staudigl, M.: Self-concordant analysis of Frank-Wolfe algorithms. *Proc. ICML* , pp. 2814–2824 (2020)

Two Motivating Papers

- ▷ Khachiyan, L.G.: Rounding of polytopes in the real number model of computation. *Mathematics of Operations Research* **21**(2), 307–320 (1996) (Elegant analysis of the FW method with exact line-search for D-optimal design)
- ▷ Dvurechensky, P., Ostroukhov, P., Safin, K., Shtern, S., Staudigl, M.: Self-concordant analysis of Frank-Wolfe algorithms. *Proc. ICML* , pp. 2814–2824 (2020)
- ▷ Dvurechensky et al. (2020) proposed and analyzed a FW method for the *whole class* of self-concordant functions. However, when specialized to D-optimal design, their complexity bound is very different from Khachiyan's result, and lacks the affine-invariance property.

Two Motivating Papers

- Khachiyan, L.G.: Rounding of polytopes in the real number model of computation. *Mathematics of Operations Research* **21**(2), 307–320 (1996) (Elegant analysis of the FW method with exact line-search for D-optimal design)

- Dvurechensky, P., Ostroukhov, P., Safin, K., Shtern, S., Staudigl, M.: Self-concordant analysis of Frank-Wolfe algorithms. *Proc. ICML* , pp. 2814–2824 (2020)

- Dvurechensky et al. (2020) proposed and analyzed a FW method for the *whole class* of self-concordant functions. However, when specialized to D-optimal design, their complexity bound is very different from Khachiyan's result, and lacks the affine-invariance property.

- We identified the *logarithmic-homogeneity* as the key element in Khachiyan's analysis, and proposed a (generalized) FW method with adaptive step-size for the much broader problem class (P).

Two Motivating Papers

- Khachiyan, L.G.: Rounding of polytopes in the real number model of computation. *Mathematics of Operations Research* **21**(2), 307–320 (1996) (Elegant analysis of the FW method with exact line-search for D-optimal design)
- Dvurechensky, P., Ostroukhov, P., Safin, K., Shtern, S., Staudigl, M.: Self-concordant analysis of Frank-Wolfe algorithms. *Proc. ICML* , pp. 2814–2824 (2020)
- Dvurechensky et al. (2020) proposed and analyzed a FW method for the *whole class* of self-concordant functions. However, when specialized to D-optimal design, their complexity bound is very different from Khachiyan's result, and lacks the affine-invariance property.
- We identified the *logarithmic-homogeneity* as the key element in Khachiyan's analysis, and proposed a (generalized) FW method with adaptive step-size for the much broader problem class (P).
- Our complexity bound essentially recovers Khachiyan's result, and is affine-invariant (along with other desirable properties).

θ-**LHSCB** (logarithmically-homogeneous self-concordant barrier)

θ-**LHSCB** (logarithmically-homogeneous self-concordant barrier)

- Let $\mathcal{K} \subsetneq \mathbb{R}^m$ be a regular cone, i.e., $\mathcal{K}$ is closed, convex, pointed and has nonempty interior.

θ-**LHSCB** (logarithmically-homogeneous self-concordant barrier)

- Let $\mathcal{K} \subsetneq \mathbb{R}^m$ be a regular cone, i.e., $\mathcal{K}$ is closed, convex, pointed and has nonempty interior.
- f is a θ-LHSCB on $\mathcal{K}$ with *complexity parameter* $\theta \geq 1$ if f is three-times differentiable and strictly convex on $\mathsf{int}\,\mathcal{K}$, and satisfies

θ-**LHSCB** (logarithmically-homogeneous self-concordant barrier)

- Let $\mathcal{K} \subsetneqq \mathbb{R}^m$ be a regular cone, i.e., $\mathcal{K}$ is closed, convex, pointed and has nonempty interior.
- f is a θ-LHSCB on $\mathcal{K}$ with *complexity parameter* $\theta \geq 1$ if f is three-times differentiable and strictly convex on $\mathsf{int}\,\mathcal{K}$, and satisfies
  1. $\left|D^3 f(u)[w,w,w]\right| \leq 2(\langle H(u)w,w\rangle)^{3/2} \;\; \forall\, u \in \mathsf{int}\,\mathcal{K}, \forall\, w \in \mathbb{R}^m$,
  2. $f(u_k) \to \infty$ for any $\{u_k\}_{k\geq 1} \subseteq \mathsf{int}\,\mathcal{K}$ such that $u_k \to u \in \mathsf{bd}\,\mathcal{K}$,
  3. $f(tu) = f(u) - \theta \ln(t) \;\; \forall\, u \in \mathsf{int}\,\mathcal{K}, \forall\, t > 0$,

  where $H(u)$ denotes the Hessian of f at $u \in \mathsf{int}\,\mathcal{K}$.

θ-**LHSCB** (logarithmically-homogeneous self-concordant barrier)

- Let $\mathcal{K} \subsetneq \mathbb{R}^m$ be a regular cone, i.e., $\mathcal{K}$ is closed, convex, pointed and has nonempty interior.

- f is a θ-LHSCB on $\mathcal{K}$ with *complexity parameter* $\theta \geq 1$ if f is three-times differentiable and strictly convex on $\mathsf{int}\,\mathcal{K}$, and satisfies

  1. $\left|D^3 f(u)[w,w,w]\right| \leq 2(\langle H(u)w,w\rangle)^{3/2} \;\; \forall\, u \in \mathsf{int}\,\mathcal{K}, \forall\, w \in \mathbb{R}^m$,
  2. $f(u_k) \to \infty$ for any $\{u_k\}_{k\geq 1} \subseteq \mathsf{int}\,\mathcal{K}$ such that $u_k \to u \in \mathsf{bd}\,\mathcal{K}$,
  3. $f(tu) = f(u) - \theta \ln(t) \;\; \forall\, u \in \mathsf{int}\,\mathcal{K}, \forall\, t > 0$,

  where $H(u)$ denotes the Hessian of f at $u \in \mathsf{int}\,\mathcal{K}$.

- Two prototypical examples:

θ-LHSCB (logarithmically-homogeneous self-concordant barrier)

- Let $\mathcal{K} \subsetneq \mathbb{R}^m$ be a regular cone, i.e., $\mathcal{K}$ is closed, convex, pointed and has nonempty interior.
- f is a θ-LHSCB on $\mathcal{K}$ with *complexity parameter* $\theta \geq 1$ if f is three-times differentiable and strictly convex on $\mathsf{int}\,\mathcal{K}$, and satisfies
  1. $\left|D^3 f(u)[w,w,w]\right| \leq 2(\langle H(u)w, w\rangle)^{3/2} \;\; \forall\, u \in \mathsf{int}\,\mathcal{K}, \forall\, w \in \mathbb{R}^m$,
  2. $f(u_k) \to \infty$ for any $\{u_k\}_{k\geq 1} \subseteq \mathsf{int}\,\mathcal{K}$ such that $u_k \to u \in \mathsf{bd}\,\mathcal{K}$,
  3. $f(tu) = f(u) - \theta \ln(t) \;\; \forall\, u \in \mathsf{int}\,\mathcal{K}, \forall\, t > 0$,

  where $H(u)$ denotes the Hessian of f at $u \in \mathsf{int}\,\mathcal{K}$.
- Two prototypical examples:
  - $f(U) = -\ln\det(U)$ for $U \in \mathcal{K} := \mathbb{S}^k_+$ and $\theta = k$,

θ-**LHSCB** (logarithmically-homogeneous self-concordant barrier)

- Let $\mathcal{K} \subsetneq \mathbb{R}^m$ be a regular cone, i.e., $\mathcal{K}$ is closed, convex, pointed and has nonempty interior.
- f is a θ-LHSCB on $\mathcal{K}$ with *complexity parameter* $\theta \geq 1$ if f is three-times differentiable and strictly convex on $\mathsf{int}\,\mathcal{K}$, and satisfies
  1. $\left|D^3 f(u)[w,w,w]\right| \leq 2(\langle H(u)w, w\rangle)^{3/2} \quad \forall\, u \in \mathsf{int}\,\mathcal{K}, \forall\, w \in \mathbb{R}^m$,
  2. $f(u_k) \to \infty$ for any $\{u_k\}_{k\geq 1} \subseteq \mathsf{int}\,\mathcal{K}$ such that $u_k \to u \in \mathsf{bd}\,\mathcal{K}$,
  3. $f(tu) = f(u) - \theta \ln(t) \quad \forall\, u \in \mathsf{int}\,\mathcal{K}, \forall\, t > 0$,

  where $H(u)$ denotes the Hessian of f at $u \in \mathsf{int}\,\mathcal{K}$.
- Two prototypical examples:
  - $f(U) = -\ln\det(U)$ for $U \in \mathcal{K} := \mathbb{S}^k_+$ and $\theta = k$,
  - $f(u) = -\sum_{j=1}^m w_j \ln(u_j)$ for $u \in \mathcal{K} := \mathbb{R}^m_+$ and $\theta = \sum_{j=1}^m w_j$ where $w_1, \ldots, w_n \geq 1$.

A Motivating Example: D-optimal Design

$$\begin{aligned} \max_p \quad & h(p) \triangleq \ln\det\left(\textstyle\sum_{i=1}^m p_i a_i a_i^\top\right) \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^m p_i = 1,\ p_i \ge 0,\ \forall\, i \in [m]. \end{aligned} \tag{D-OPT}$$

A Motivating Example: D-optimal Design

$$\begin{aligned} \max_p \;\; & h(p) \triangleq \ln\det\left(\textstyle\sum_{i=1}^m p_i a_i a_i^\top\right) \\ \text{s.t.} \;\; & \textstyle\sum_{i=1}^m p_i = 1,\; p_i \geq 0,\; \forall\, i \in [m]. \end{aligned} \tag{D-OPT}$$

- Problem data: $\{a_i\}_{i=1}^m \subseteq \mathbb{R}^n$.

A Motivating Example: D-optimal Design

$$\begin{aligned} \max_p \;\; & h(p) \triangleq \ln\det\left(\textstyle\sum_{i=1}^m p_i a_i a_i^\top\right) \\ \text{s.t.} \;\; & \textstyle\sum_{i=1}^m p_i = 1,\; p_i \ge 0,\; \forall\, i \in [m]. \end{aligned} \tag{D-OPT}$$

- Problem data: $\{a_i\}_{i=1}^m \subseteq \mathbb{R}^n$.
- Arises in many places, including optimal experimental design, and as the dual problem of the minimum volume enclosing ellipsoid (MVEE) problem.

A Motivating Example: D-optimal Design

$$\begin{aligned} \max_p \;\; & h(p) \triangleq \ln\det\left(\textstyle\sum_{i=1}^m p_i a_i a_i^\top\right) \\ \text{s.t.} \;\; & \textstyle\sum_{i=1}^m p_i = 1,\; p_i \ge 0,\; \forall\, i \in [m]. \end{aligned} \tag{D-OPT}$$

- Problem data: $\{a_i\}_{i=1}^m \subseteq \mathbb{R}^n$.
- Arises in many places, including optimal experimental design, and as the dual problem of the minimum volume enclosing ellipsoid (MVEE) problem.
- Khachiyan (1996) proposed a "barycentric coordinate ascent" method with exact line-search, which is actually FW with exact line-search. Method works remarkably well both in theory and practice: it computes an ε-optimal solution of (D-OPT) in (essentially) $O(n^2/\varepsilon)$ iterations.

A Motivating Example: D-optimal Design

$$\begin{aligned} \max_p \quad & h(p) \triangleq \ln\det\left(\textstyle\sum_{i=1}^m p_i a_i a_i^\top\right) \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^m p_i = 1,\ p_i \ge 0,\ \forall\, i \in [m]. \end{aligned} \tag{D-OPT}$$

- Problem data: $\{a_i\}_{i=1}^m \subseteq \mathbb{R}^n$.
- Arises in many places, including optimal experimental design, and as the dual problem of the minimum volume enclosing ellipsoid (MVEE) problem.
- Khachiyan (1996) proposed a "barycentric coordinate ascent" method with exact line-search, which is actually FW with exact line-search. Method works remarkably well both in theory and practice: it computes an ε-optimal solution of (D-OPT) in (essentially) $O(n^2/\varepsilon)$ iterations.
- The theoretical success of this method has been a mysterious outlier for more than 20 years, since (D-OPT) does not satisfy the usual L-smooth curvature condition in (LSm). What problem structure actually drives the complexity bound? And might such structure exist anywhere else?

A Motivating Example: D-optimal Design

$$\begin{aligned} \max_p \quad & h(p) \triangleq \ln\det\left(\textstyle\sum_{i=1}^m p_i a_i a_i^\top\right) \\ \text{s. t.} \quad & \textstyle\sum_{i=1}^m p_i = 1,\ p_i \geq 0,\ \forall\, i \in [m]. \end{aligned} \tag{D-OPT}$$

- Problem data: $\{a_i\}_{i=1}^m \subseteq \mathbb{R}^n$.
- Arises in many places, including optimal experimental design, and as the dual problem of the minimum volume enclosing ellipsoid (MVEE) problem.
- Khachiyan (1996) proposed a "barycentric coordinate ascent" method with exact line-search, which is actually FW with exact line-search. Method works remarkably well both in theory and practice: it computes an ε-optimal solution of (D-OPT) in (essentially) $O(n^2/\varepsilon)$ iterations.
- The theoretical success of this method has been a mysterious outlier for more than 20 years, since (D-OPT) does not satisfy the usual L-smooth curvature condition in (LSm). What problem structure actually drives the complexity bound? And might such structure exist anywhere else?
- We resolve this mystery and generalize his method to the much broader class of problems in (P), even while relaxing the exact line-search requirement.

Another Example: Poisson Image Deblurring with TV Regularization

Another Example: Poisson Image Deblurring with TV Regularization

- Let an $m \times n$ matrix X denote the true representation of an image, such that $0 \leq X_{ij} \leq M$ denotes the pixel level at location (i,j).

Another Example: Poisson Image Deblurring with TV Regularization

- Let an $m \times n$ matrix X denote the true representation of an image, such that $0 \leq X_{ij} \leq M$ denotes the pixel level at location (i, j).
- Let $\mathsf{A} : \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n}$ denote the 2D discrete convolutional (linear) operator, which is assumed to be known.

Another Example: Poisson Image Deblurring with TV Regularization

- Let an $m \times n$ matrix X denote the true representation of an image, such that $0 \leq X_{ij} \leq M$ denotes the pixel level at location (i, j).
- Let $\mathsf{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ denote the 2D discrete convolutional (linear) operator, which is assumed to be known.
- The observed image Y is obtained by first passing X through A, and then is assumed to be subject to additive independent (entry-wise) Poisson noise.

Another Example: Poisson Image Deblurring with TV Regularization

- Let an $m \times n$ matrix X denote the true representation of an image, such that $0 \leq X_{ij} \leq M$ denotes the pixel level at location (i, j).
- Let $\mathsf{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ denote the 2D discrete convolutional (linear) operator, which is assumed to be known.
- The observed image Y is obtained by first passing X through A, and then is assumed to be subject to additive independent (entry-wise) Poisson noise.
- For convenience, we also represent A in its matrix form $A \in \mathbb{R}^{N\times N}$, where $N := mn$, and vectorize Y and X into $y \in \mathbb{R}^N$ and $x \in \mathbb{R}^N$, respectively. Notation: we write $x = \mathsf{vec}(X)$ and $X = \mathsf{mat}(x)$, etc.

Poisson Image Deblurring with TV Regularization, continued

Poisson Image Deblurring with TV Regularization, continued

- We seek to recover X from Y (equivalently x from y) using maximum-likelihood estimation on the TV-regularized problem:

$$\begin{aligned} \min_{x\in\mathbb{R}^N} \quad & \bar{F}(x) := -\sum_{l=1}^{N} y_l \ln(a_l^\top x) + \left(\sum_{l=1}^{N} a_l\right)^\top x + \lambda \mathrm{TV}(x) \\ \text{s.t.} \quad & 0 \le x \le Me \ , \end{aligned} \qquad (\texttt{Deblur})$$

Poisson Image Deblurring with TV Regularization, continued

- We seek to recover X from Y (equivalently x from y) using maximum-likelihood estimation on the TV-regularized problem:

$$\begin{aligned} \min_{x\in\mathbb{R}^N} \quad & \bar{F}(x) := -\textstyle\sum_{l=1}^N y_l \ln(a_l^\top x) + (\sum_{l=1}^N a_l)^\top x + \lambda \mathrm{TV}(x) \\ \mathrm{s.\,t.} \quad & 0 \le x \le Me \ , \end{aligned} \qquad \texttt{(Deblur)}$$

- (`Deblur`) has a (standard) total-variation (TV) regularization term to recover a smooth image with sharp edges. The TV term is given by

$$\begin{aligned} \mathrm{TV}(x) := & \textstyle\sum_{i=1}^m \sum_{j=1}^{n-1} \left| [\mathsf{mat}(x)]_{i,j} - [\mathsf{mat}(x)]_{i,j+1} \right| \\ & + \textstyle\sum_{i=1}^{m-1} \sum_{j=1}^{n} \left| [\mathsf{mat}(x)]_{i,j} - [\mathsf{mat}(x)]_{i+1,j} \right| . \end{aligned}$$

Some Other Applications

Some Other Applications

- Positron emission tomography (PET)

Some Other Applications

- Positron emission tomography (PET)
- Optimal expected log investment (Cover (1984))

Some Other Applications

- Positron emission tomography (PET)
- Optimal expected log investment (Cover (1984))
- Computation of the analytic center of a polytope

Our Method: (generalized) Frank-Wolfe (gFW-LHSCB)

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \tag{P}$$

Our Method: (generalized) Frank-Wolfe (gFW-LHSCB)

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \tag{P}$$

- **Initialize**: $x^0 \in \mathsf{dom}\, F$, $k := 0$

Our Method: (generalized) Frank-Wolfe (gFW-LHSCB)

$$F^* := \min_{x\in\mathbb{R}^n} \; [F(x) := f(\mathsf{A}x) + h(x)] \qquad \text{(P)}$$

- **Initialize**: $x^0 \in \mathsf{dom}\, F$, $k := 0$
- **Repeat** (until some convergence criterion is met)

$$v^k \in \arg\min_{x\in\mathbb{R}^n} \langle \nabla f(\mathsf{A}x^k), \mathsf{A}x\rangle + h(x) \qquad \text{(Solve Lin. subproblem)}$$

Our Method: (generalized) Frank-Wolfe (gFW-LHSCB)

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \qquad \texttt{(P)}$$

- **Initialize**: $x^0 \in \mathsf{dom}\, F$, $k := 0$
- **Repeat** (until some convergence criterion is met)

$$v^k \in \arg\min_{x\in\mathbb{R}^n} \langle \nabla f(\mathsf{A}x^k), \mathsf{A}x\rangle + h(x) \qquad \text{(Solve Lin. subproblem)}$$

$$G_k := \langle \nabla f(\mathsf{A}x^k), \mathsf{A}(x^k - v^k)\rangle + h(x^k) - h(v^k) \qquad \text{(FW Gap)}$$

Our Method: (generalized) Frank-Wolfe (gFW-LHSCB)

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \tag{P}$$

- **Initialize**: $x^0 \in \mathsf{dom}\, F$, $k := 0$
- **Repeat** (until some convergence criterion is met)

$$v^k \in \arg\min_{x\in\mathbb{R}^n} \langle \nabla f(\mathsf{A}x^k), \mathsf{A}x\rangle + h(x) \qquad \text{(Solve Lin. subproblem)}$$

$$G_k := \langle \nabla f(\mathsf{A}x^k), \mathsf{A}(x^k - v^k)\rangle + h(x^k) - h(v^k) \qquad \text{(FW Gap)}$$

$$D_k := D_k := \|\mathsf{A}(v^k - x^k)\|_{\mathsf{A}x^k} \qquad \text{(Local Distance)}$$

Our Method: (generalized) Frank-Wolfe (gFW-LHSCB)

$$F^* := \min_{x\in\mathbb{R}^n} \; [F(x) := f(\mathsf{A}x) + h(x)] \tag{P}$$

- **Initialize**: $x^0 \in \mathsf{dom}\, F$, $k := 0$
- **Repeat** (until some convergence criterion is met)

$$v^k \in \arg\min_{x\in\mathbb{R}^n} \langle \nabla f(\mathsf{A}x^k), \mathsf{A}x\rangle + h(x) \qquad \text{(Solve Lin. subproblem)}$$

$$G_k := \langle \nabla f(\mathsf{A}x^k), \mathsf{A}(x^k - v^k)\rangle + h(x^k) - h(v^k) \qquad \text{(FW Gap)}$$

$$D_k := D_k := \|\mathsf{A}(v^k - x^k)\|_{\mathsf{A}x^k} \qquad \text{(Local Distance)}$$

$$\alpha_k := \min\left\{ \frac{G_k}{D_k(G_k + D_k)} \, , 1 \right\} \qquad \text{(Stepsize)}$$

Our Method: (generalized) Frank-Wolfe (gFW-LHSCB)

$$F^* := \min_{x\in\mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x)\right] \tag{P}$$

- **Initialize**: $x^0 \in \mathsf{dom}\, F$, $k := 0$
- **Repeat** (until some convergence criterion is met)

$$v^k \in \arg\min_{x\in\mathbb{R}^n} \langle \nabla f(\mathsf{A}x^k), \mathsf{A}x\rangle + h(x) \qquad \text{(Solve Lin. subproblem)}$$

$$G_k := \langle \nabla f(\mathsf{A}x^k), \mathsf{A}(x^k - v^k)\rangle + h(x^k) - h(v^k) \qquad \text{(FW Gap)}$$

$$D_k := D_k := \|\mathsf{A}(v^k - x^k)\|_{\mathsf{A}x^k} \qquad \text{(Local Distance)}$$

$$\alpha_k := \min\left\{\frac{G_k}{D_k(G_k + D_k)}, 1\right\} \qquad \text{(Stepsize)}$$

$$x^{k+1} := x^k + \alpha_k(v^k - x^k) \qquad \text{(Update)}$$

Our Method: (generalized) Frank-Wolfe (gFW-LHSCB)

$$F^* := \min_{x \in \mathbb{R}^n} \left[F(x) := f(\mathsf{A}x) + h(x) \right] \tag{P}$$

- **Initialize**: $x^0 \in \mathsf{dom}\, F$, $k := 0$
- **Repeat** (until some convergence criterion is met)

$$v^k \in \arg\min_{x \in \mathbb{R}^n} \langle \nabla f(\mathsf{A}x^k), \mathsf{A}x \rangle + h(x) \qquad \text{(Solve Lin. subproblem)}$$

$$G_k := \langle \nabla f(\mathsf{A}x^k), \mathsf{A}(x^k - v^k) \rangle + h(x^k) - h(v^k) \qquad \text{(FW Gap)}$$

$$D_k := D_k := \|\mathsf{A}(v^k - x^k)\|_{\mathsf{A}x^k} \qquad \text{(Local Distance)}$$

$$\alpha_k := \min \left\{ \frac{G_k}{D_k(G_k + D_k)}, 1 \right\} \qquad \text{(Stepsize)}$$

$$x^{k+1} := x^k + \alpha_k (v^k - x^k) \qquad \text{(Update)}$$

$$k := k + 1$$

Remarks on gFW-LHSCB

Remarks on gFW-LHSCB

- When h is the indicator function $h = \iota_{\mathcal{X}}$, then gFW-LHSCB specializes exactly to the algorithm of Dvurechensky et al. (2020).

Remarks on gFW-LHSCB

- When h is the indicator function $h = \iota_{\mathcal{X}}$, then gFW-LHSCB specializes exactly to the algorithm of Dvurechensky et al. (2020).
- For most applications (including all of the applications mentioned previously), D_k in (Local Distance) can be computed in $O(n)$ time.

Remarks on gFW-LHSCB

- When h is the indicator function $h = \iota_{\mathcal{X}}$, then gFW-LHSCB specializes exactly to the algorithm of Dvurechensky et al. (2020).
- For most applications (including all of the applications mentioned previously), D_k in (Local Distance) can be computed in $O(n)$ time.
- The step-size rule in (Stepsize) is derived from the "curvature property" of a (standard) self-concordant function:

$$f(x^k + \alpha(v^k - x^k)) \leq f(x^k) - \alpha G_k + \omega(\alpha D_k), \qquad \texttt{(Curvature)}$$

where $\omega(t) := -t - \ln(1-t)$ for $t < 1$.

Remarks on gFW-LHSCB

- When h is the indicator function $h = \iota_{\mathcal{X}}$, then gFW-LHSCB specializes exactly to the algorithm of Dvurechensky et al. (2020).
- For most applications (including all of the applications mentioned previously), D_k in (Local Distance) can be computed in $O(n)$ time.
- The step-size rule in (Stepsize) is derived from the "curvature property" of a (standard) self-concordant function:

$$f(x^k + \alpha(v^k - x^k)) \le f(x^k) - \alpha G_k + \omega(\alpha D_k), \qquad \texttt{(Curvature)}$$

  where $\omega(t) := -t - \ln(1-t)$ for $t < 1$.
- Neither the algorithm nor (`Curvature`) use the special properties of the barrier or the logarithmic homogeneity of f. However, these properties drive our complexity analysis.

Computational Guarantees

Define $\delta_k := F(x^k) - F^*$ for $k \geq 0$ (hence δ_0 is the initial optimality gap)

Define $R_h := \max_{x,y \in \mathsf{dom}\, h} \; |h(x) - h(y)|$ (the variation of h on its domain)

Theorem:

Computational Guarantees

Define $\delta_k := F(x^k) - F^*$ for $k \geq 0$ (hence δ_0 is the initial optimality gap)

Define $R_h := \max_{x,y \in \mathsf{dom}\, h} \; |h(x) - h(y)|$ (the variation of h on its domain)

Theorem:

▷ (Iteration complexity for ε-optimality gap) Let K_ε denote the number of iterations required by gFW-LHSCB to obtain $\delta_k \leq \varepsilon$. Then:

$$K_\varepsilon \leq \lceil 5.3(\delta_0 + \theta + R_h) \ln(10.6\delta_0) \rceil + \left\lceil 12(\theta + R_h)^2 \max\left\{ \frac{1}{\varepsilon} - \frac{1}{\delta_0} \;, 0 \right\} \right\rceil \; .$$

Computational Guarantees

Define $\delta_k := F(x^k) - F^*$ for $k \geq 0$ (hence δ_0 is the initial optimality gap)

Define $R_h := \max_{x,y \in \mathsf{dom}\, h} \; |h(x) - h(y)|$ (the variation of h on its domain)

Theorem:

- (Iteration complexity for ε-optimality gap) Let K_ε denote the number of iterations required by gFW-LHSCB to obtain $\delta_k \leq \varepsilon$. Then:

$$K_\varepsilon \leq \lceil 5.3(\delta_0 + \theta + R_h) \ln(10.6\delta_0) \rceil + \left\lceil 12(\theta + R_h)^2 \max\left\{ \frac{1}{\varepsilon} - \frac{1}{\delta_0}, 0 \right\} \right\rceil .$$

- (Iteration complexity for ε-FW gap) Let $\mathrm{FWGAP}_\varepsilon$ denote the number of iterations required by gFW-LHSCB to obtain $G_k \leq \varepsilon$. Then:

$$\mathrm{FWGAP}_\varepsilon \leq \lceil 5.3(\delta_0 + \theta + R_h) \ln(10.6\delta_0) \rceil + \left\lceil \frac{24(\theta + R_h)^2}{\varepsilon} \right\rceil .$$

Remarks on the Computational Guarantees

Remarks on the Computational Guarantees

- Our computational guarantees only depend on three (natural) quantities:

Remarks on the Computational Guarantees

- Our computational guarantees only depend on three (natural) quantities:
  - the initial optimality gap δ_0,

Remarks on the Computational Guarantees

- Our computational guarantees only depend on three (natural) quantities:
  - the initial optimality gap δ_0,
  - the complexity parameter θ of the barrier f,

Remarks on the Computational Guarantees

▷ Our computational guarantees only depend on three (natural) quantities:

- the initial optimality gap δ_0,
- the complexity parameter θ of the barrier f,
- the variation of h on its domain $\mathsf{dom}\, h$ ($= 0$ if $h = \iota_{\mathcal{X}}$).

Remarks on the Computational Guarantees

- Our computational guarantees only depend on three (natural) quantities:
  - the initial optimality gap δ_0,
  - the complexity parameter θ of the barrier f,
  - the variation of h on its domain $\mathsf{dom}\, h$ ($= 0$ if $h = \iota_{\mathcal{X}}$).
- Comparison with Khachiyan's results for (D-OPT):

Remarks on the Computational Guarantees

- Our computational guarantees only depend on three (natural) quantities:
  - the initial optimality gap δ_0,
  - the complexity parameter θ of the barrier f,
  - the variation of h on its domain $\mathsf{dom}\, h$ ($= 0$ if $h = \iota_{\mathcal{X}}$).
- Comparison with Khachiyan's results for (D-OPT):
  - In (D-OPT), we have $\theta = n$, $R_h = 0$, and if $x^0 = (1/m)e$, then $\delta_0 \leq n \ln(m/n)$.

Remarks on the Computational Guarantees

▷ Our computational guarantees only depend on three (natural) quantities:

- the initial optimality gap δ_0,
- the complexity parameter θ of the barrier f,
- the variation of h on its domain $\mathsf{dom}\, h$ ($= 0$ if $h = \iota_{\mathcal{X}}$).

▷ Comparison with Khachiyan's results for (`D-OPT`):

- In (`D-OPT`), we have $\theta = n$, $R_h = 0$, and if $x^0 = (1/m)e$, then $\delta_0 \le n \ln(m/n)$.
- Using the adaptive step-size, our complexity bound specializes to

$$O\Big(n \ln(m/n)(\ln n + \ln\ln(m/n)) + n^2/\varepsilon\Big)\ . \qquad \texttt{(Ours)}$$

Remarks on the Computational Guarantees

- Our computational guarantees only depend on three (natural) quantities:
  - the initial optimality gap δ_0,
  - the complexity parameter θ of the barrier f,
  - the variation of h on its domain $\mathsf{dom}\, h$ ($= 0$ if $h = \iota_{\mathcal{X}}$).
- Comparison with Khachiyan's results for (`D-OPT`):
  - In (`D-OPT`), we have $\theta = n$, $R_h = 0$, and if $x^0 = (1/m)e$, then $\delta_0 \le n \ln(m/n)$.
  - Using the adaptive step-size, our complexity bound specializes to

$$O\Big(n \ln(m/n)(\ln n + \ln\ln(m/n)) + n^2/\varepsilon\Big) \ . \qquad \texttt{(Ours)}$$

  - Using exact line-search, Khachiyan's bound is

$$O\Big(n(\ln n + \ln\ln(m/n)) + n^2/\varepsilon\Big) \ . \qquad \texttt{(Kha)}$$

Remarks on the Computational Guarantees

- Our computational guarantees only depend on three (natural) quantities:
  - the initial optimality gap δ_0,
  - the complexity parameter θ of the barrier f,
  - the variation of h on its domain $\mathsf{dom}\, h$ $(= 0$ if $h = \iota_{\mathcal{X}})$.
- Comparison with Khachiyan's results for (D-OPT):
  - In (D-OPT), we have $\theta = n$, $R_h = 0$, and if $x^0 = (1/m)e$, then $\delta_0 \le n \ln(m/n)$.
  - Using the adaptive step-size, our complexity bound specializes to
    $$O\Big(n \ln(m/n)(\ln n + \ln\ln(m/n)) + n^2/\varepsilon\Big)\ . \qquad \texttt{(Ours)}$$
  - Using exact line-search, Khachiyan's bound is
    $$O\Big(n(\ln n + \ln\ln(m/n)) + n^2/\varepsilon\Big)\ . \qquad \texttt{(Kha)}$$
  - Observe that (Ours) has the exact same dependence on ε as (Kha), namely $O(n^2/\varepsilon)$, but the "fixed" term is slightly inferior to (Kha) by the factor $O(\ln(m/n))$.

Computational Experiments on Poisson Image Deblurring with TV Regularization (Deblur)

$$\min_{x\in\mathbb{R}^N} \quad \bar{F}(x) := \underbrace{-\sum_{l=1}^N y_l \ln(a_l^\top x)}_{=f(\mathsf{A}x)} + \underbrace{\langle \sum_{l=1}^N a_l, x\rangle + \lambda \mathrm{TV}(x)}_{=h(x)}$$

$$\text{s.t.} \quad 0 \le x \le Me \;, \qquad (\texttt{Deblur})$$

Computational Experiments on Poisson Image Deblurring with TV Regularization (`Deblur`)

$$\min_{x\in\mathbb{R}^N} \quad \bar{F}(x) := \underbrace{-\textstyle\sum_{l=1}^N y_l \ln(a_l^\top x)}_{=f(\mathsf{A}x)} + \underbrace{\langle \textstyle\sum_{l=1}^N a_l, x\rangle + \lambda \mathrm{TV}(x)}_{=h(x)}$$

$$\text{s.t.} \quad 0 \le x \le Me \;, \qquad \texttt{(Deblur)}$$

- Very few principled first-order methods have been proposed to solve (`Deblur`), because:

Computational Experiments on Poisson Image Deblurring with TV Regularization (Deblur)

$$\min_{x\in\mathbb{R}^N} \;\; \bar{F}(x) := \underbrace{-\textstyle\sum_{l=1}^N y_l \ln(a_l^\top x)}_{=f(\mathsf{A}x)} + \underbrace{\langle \textstyle\sum_{l=1}^N a_l, x\rangle + \lambda \mathrm{TV}(x)}_{=h(x)}$$

$$\text{s.t.} \quad 0 \le x \le Me \;, \qquad \texttt{(Deblur)}$$

▷ Very few principled first-order methods have been proposed to solve (Deblur), because:

- $f : u \mapsto -\sum_{l=1}^N y_l \ln(u_l)$ is neither Lipschitz nor L-smooth on the set $\{u \in \mathbb{R}^N : u = \mathsf{A}x,\ 0 \le x \le Me\}$, and

Computational Experiments on Poisson Image Deblurring with TV Regularization (Deblur)

$$\min_{x\in\mathbb{R}^N} \quad \bar{F}(x) := \underbrace{-\sum_{l=1}^N y_l \ln(a_l^\top x)}_{=f(\mathsf{A}x)} + \underbrace{\langle \sum_{l=1}^N a_l, x\rangle + \lambda \mathrm{TV}(x)}_{=h(x)}$$

$$\text{s.t.} \quad 0 \le x \le Me\ , \qquad \texttt{(Deblur)}$$

- Very few principled first-order methods have been proposed to solve (Deblur), because:
  - $f : u \mapsto -\sum_{l=1}^N y_l \ln(u_l)$ is neither Lipschitz nor L-smooth on the set $\{u \in \mathbb{R}^N : u = \mathsf{A}x,\ 0 \le x \le Me\}$, and
  - $\mathrm{TV}(\cdot)$ does not have an efficiently computable proximal operator.

Computational Experiments on Poisson Image Deblurring with TV Regularization (Deblur)

$$\min_{x\in\mathbb{R}^N} \quad \bar{F}(x) := \underbrace{-\textstyle\sum_{l=1}^N y_l \ln(a_l^\top x)}_{=f(\mathsf{A}x)} + \underbrace{\langle \textstyle\sum_{l=1}^N a_l, x\rangle + \lambda \mathrm{TV}(x)}_{=h(x)}$$
$$\text{s.t.} \quad 0 \le x \le Me \;, \qquad \texttt{(Deblur)}$$

- ▷ Very few principled first-order methods have been proposed to solve (Deblur), because:
  - $f : u \mapsto -\sum_{l=1}^N y_l \ln(u_l)$ is neither Lipschitz nor L-smooth on the set $\{u \in \mathbb{R}^N : u = \mathsf{A}x,\ 0 \le x \le Me\}$, and
  - $\mathrm{TV}(\cdot)$ does not have an efficiently computable proximal operator.
- ▷ However, $\mathrm{TV}(\cdot)$ is a polyhedral function, and the linear-optimization sub-problem

$$v^k \in \arg\min_{0\le x\le Me} \langle \nabla f(\mathsf{A}x^k), \mathsf{A}x\rangle + \langle \textstyle\sum_{l=1}^N a_l, x\rangle + \lambda \mathrm{TV}(x)$$

can be formulated as a relatively simple LP and solved easily using a standard LP solver such as Gurobi.

Implementation Details/Issues

Implementation Details/Issues

- We evaluate the numerical performance of our FW method gFW-LHSCB (with adaptive stepsize) which we call `FW-Adapt`.

Implementation Details/Issues

- We evaluate the numerical performance of our FW method gFW-LHSCB (with adaptive stepsize) which we call `FW-Adapt`.
- It turns out that an exact line-search step-size for gFW-LHSCB can be computed for this particular problem, which we call `FW-Exact`.

Implementation Details/Issues

- We evaluate the numerical performance of our FW method gFW-LHSCB (with adaptive stepsize) which we call `FW-Adapt`.
- It turns out that an exact line-search step-size for gFW-LHSCB can be computed for this particular problem, which we call `FW-Exact`.
- We tested `FW-Adapt` and `FW-Exact` on the Shepp-Logan phantom image of size 100×100 (hence $N = 10,000$).

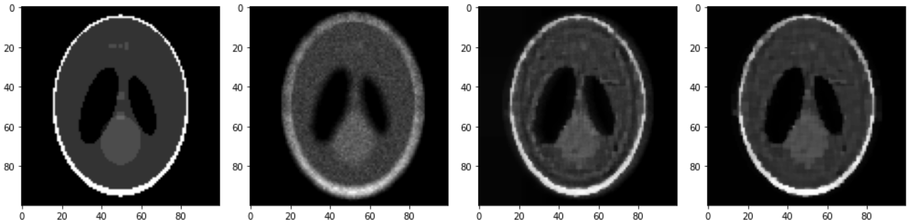
Implementation Details/Issues

- ▷ We evaluate the numerical performance of our FW method gFW-LHSCB (with adaptive stepsize) which we call `FW-Adapt`.
- ▷ It turns out that an exact line-search step-size for gFW-LHSCB can be computed for this particular problem, which we call `FW-Exact`.
- ▷ We tested `FW-Adapt` and `FW-Exact` on the Shepp-Logan phantom image of size 100×100 (hence $N = 10,000$).
- ▷ We chose the starting point $x^0 = \mathsf{vec}(Y)$, and we set $\lambda = 0.01$.

Implementation Details/Issues

- We evaluate the numerical performance of our FW method gFW-LHSCB (with adaptive stepsize) which we call `FW-Adapt`.
- It turns out that an exact line-search step-size for gFW-LHSCB can be computed for this particular problem, which we call `FW-Exact`.
- We tested `FW-Adapt` and `FW-Exact` on the Shepp-Logan phantom image of size 100×100 (hence $N = 10,000$).
- We chose the starting point $x^0 = \mathsf{vec}(Y)$, and we set $\lambda = 0.01$.
- We used CVXPY to (approximately) compute the optimal objective value $\bar{F}^*$ of (`Deblur`) in order to compute optimality gaps.

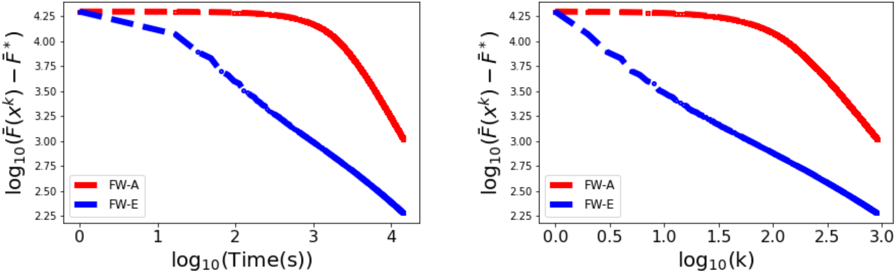
Results: Recovered Images



(a) True image X (b) Noisy image Y (c) `FW-Adapt` (d) `FW-Exact`

Figure 1:
True, noisy and recovered Shepp-Logan phantom image.

Results: Optimality Gaps versus Time and Iterations



(a) Optimality gap versus time (in seconds)

(b) Optimality gap versus iterations

Figure 2:
Comparison of empirical optimality gaps of `FW-Adapt` (`FW-A`) and `FW-Exact` (`FW-E`) for image recovery of the Shepp-Logan phantom image.

Thank you!

Comparison with Dvurechensky et al. (2020)

Comparison with Dvurechensky et al. (2020)

- To find an ε-optimal solution, the complexity bound in Dvurechensky et al. (2020) reads:

$$O\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\ln\left(\delta_0/\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\right)\right)+L(x^0)D^2_{\mathcal{X},\|\cdot\|_2}/\varepsilon\right), \qquad \texttt{(Dvu)}$$

where $\mathcal{S}(x^0) := \{x \in \mathsf{dom}\, F \cap \mathcal{X} \,:\, F(x) \le F(x^0)\}$ denotes the initial level-set and

$$L(x^0) := \max_{x\in\mathcal{S}(x^0)} \|\nabla^2 \bar{F}(x)\|_2 < +\infty \text{ , and } D_{\mathcal{X},\|\cdot\|_2} := \max_{x,y\in\mathcal{X}} \|x-y\|_2 \quad .$$

Comparison with Dvurechensky et al. (2020)

- To find an ε-optimal solution, the complexity bound in Dvurechensky et al. (2020) reads:

$$O\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\ln\left(\delta_0/\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\right)\right)+L(x^0)D^2_{\mathcal{X},\|\cdot\|_2}/\varepsilon\right), \qquad \texttt{(Dvu)}$$

where $\mathcal{S}(x^0) := \{x \in \mathsf{dom}\, F \cap \mathcal{X} \,:\, F(x) \le F(x^0)\}$ denotes the initial level-set and

$$L(x^0) := \max_{x\in\mathcal{S}(x^0)} \|\nabla^2 \bar{F}(x)\|_2 < +\infty \text{ , and } D_{\mathcal{X},\|\cdot\|_2} := \max_{x,y\in\mathcal{X}} \|x-y\|_2 \quad .$$

- Specialized to the traditional setting, our complexity bound reads:

$$O\Big((\delta_0+\theta)\ln(\delta_0)+(\theta)^2/\varepsilon\Big). \qquad \texttt{(Ours)}$$

Comparison with Dvurechensky et al. (2020)

- To find an ε-optimal solution, the complexity bound in Dvurechensky et al. (2020) reads:

$$O\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\ln\left(\delta_0/\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\right)\right) + L(x^0)D^2_{\mathcal{X},\|\cdot\|_2}/\varepsilon\right), \qquad \texttt{(Dvu)}$$

where $\mathcal{S}(x^0) := \{x \in \mathsf{dom}\, F \cap \mathcal{X} \,:\, F(x) \le F(x^0)\}$ denotes the initial level-set and

$$L(x^0) := \max_{x\in\mathcal{S}(x^0)} \|\nabla^2 \bar{F}(x)\|_2 < +\infty \text{ , and } D_{\mathcal{X},\|\cdot\|_2} := \max_{x,y\in\mathcal{X}} \|x-y\|_2 \quad .$$

- Specialized to the traditional setting, our complexity bound reads:

$$O\Big((\delta_0+\theta)\ln(\delta_0) + (\theta)^2/\varepsilon\Big). \qquad \texttt{(Ours)}$$

- Our bound (Ours) has the following merits:

Comparison with Dvurechensky et al. (2020)

- To find an ε-optimal solution, the complexity bound in Dvurechensky et al. (2020) reads:

$$O\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\ln\left(\delta_0/\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\right)\right)+L(x^0)D^2_{\mathcal{X},\|\cdot\|_2}/\varepsilon\right), \qquad \texttt{(Dvu)}$$

where $\mathcal{S}(x^0) := \{x \in \mathsf{dom}\, F \cap \mathcal{X} \,:\, F(x) \le F(x^0)\}$ denotes the initial level-set and

$$L(x^0) := \max_{x\in\mathcal{S}(x^0)} \|\nabla^2 \bar{F}(x)\|_2 < +\infty \text{ , and } D_{\mathcal{X},\|\cdot\|_2} := \max_{x,y\in\mathcal{X}} \|x-y\|_2 \quad .$$

- Specialized to the traditional setting, our complexity bound reads:

$$O\Big((\delta_0+\theta)\ln(\delta_0)+(\theta)^2/\varepsilon\Big). \qquad \texttt{(Ours)}$$

- Our bound (Ours) has the following merits:
  - Affine-invariance

Comparison with Dvurechensky et al. (2020)

- To find an ε-optimal solution, the complexity bound in Dvurechensky et al. (2020) reads:

$$O\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\ln\left(\delta_0/\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\right)\right) + L(x^0)D^2_{\mathcal{X},\|\cdot\|_2}/\varepsilon\right), \qquad \texttt{(Dvu)}$$

where $\mathcal{S}(x^0) := \{x \in \mathsf{dom}\, F \cap \mathcal{X} \,:\, F(x) \le F(x^0)\}$ denotes the initial level-set and

$$L(x^0) := \max_{x\in\mathcal{S}(x^0)} \|\nabla^2 \bar{F}(x)\|_2 < +\infty\ , \text{and}\ \ D_{\mathcal{X},\|\cdot\|_2} := \max_{x,y\in\mathcal{X}} \|x-y\|_2 \quad .$$

- Specialized to the traditional setting, our complexity bound reads:

$$O\big((\delta_0+\theta)\ln(\delta_0) + (\theta)^2/\varepsilon\big). \qquad \texttt{(Ours)}$$

- Our bound (**Ours**) has the following merits:
  - Affine-invariance
  - Norm-invariance

Comparison with Dvurechensky et al. (2020)

- To find an ε-optimal solution, the complexity bound in Dvurechensky et al. (2020) reads:

$$O\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\ln\left(\delta_0/\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\right)\right)+L(x^0)D^2_{\mathcal{X},\|\cdot\|_2}/\varepsilon\right), \qquad \texttt{(Dvu)}$$

where $\mathcal{S}(x^0) := \{x \in \mathsf{dom}\, F \cap \mathcal{X} \,:\, F(x) \le F(x^0)\}$ denotes the initial level-set and

$$L(x^0) := \max_{x\in\mathcal{S}(x^0)} \|\nabla^2 \bar{F}(x)\|_2 < +\infty \,, \text{and} \;\; D_{\mathcal{X},\|\cdot\|_2} := \max_{x,y\in\mathcal{X}} \|x-y\|_2 \quad .$$

- Specialized to the traditional setting, our complexity bound reads:

$$O\big((\delta_0+\theta)\ln(\delta_0)+(\theta)^2/\varepsilon\big). \qquad \texttt{(Ours)}$$

- Our bound (**Ours**) has the following merits:
  - Affine-invariance
  - Norm-invariance
  - Interpretability

Comparison with Dvurechensky et al. (2020)

- To find an ε-optimal solution, the complexity bound in Dvurechensky et al. (2020) reads:

$$O\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\ln\left(\delta_0/\left(\sqrt{L(x^0)}D_{\mathcal{X},\|\cdot\|_2}\right)\right)+L(x^0)D^2_{\mathcal{X},\|\cdot\|_2}/\varepsilon\right), \qquad \texttt{(Dvu)}$$

where $\mathcal{S}(x^0) := \{x \in \mathsf{dom}\, F \cap \mathcal{X} \,:\, F(x) \le F(x^0)\}$ denotes the initial level-set and

$$L(x^0) := \max_{x\in\mathcal{S}(x^0)} \|\nabla^2 \bar{F}(x)\|_2 < +\infty \text{ , and } D_{\mathcal{X},\|\cdot\|_2} := \max_{x,y\in\mathcal{X}} \|x-y\|_2 \quad .$$

- Specialized to the traditional setting, our complexity bound reads:

$$O\Big((\delta_0+\theta)\ln(\delta_0)+(\theta)^2/\varepsilon\Big). \qquad \texttt{(Ours)}$$

- Our bound (Ours) has the following merits:
  - Affine-invariance
  - Norm-invariance
  - Interpretability
  - Ease of parameter estimation