Mathematics of Data: From Theory to Computation

Prof. Volkan Cevher
volkan.cevher@epfl.ch

Lecture 11: Constrained convex minimization II

Laboratory for Information and Inference Systems (LIONS)
École Polytechnique Fédérale de Lausanne (EPFL)

EE-556 (Fall 2017)

License Information for Mathematics of Data Slides

Outline

- This class:
  1. Frank-Wolfe method
  2. Universal primal-dual gradient methods
  3. ADMM
- Next class
  1. Disciplined convex programming

Recommended reading material

- M. Jaggi, *Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization* In Proc. 30th International Conference on Machine Learning, 2013.
- A. Yurtsever, Q. Tran-Dinh and V. Cevher, *A Universal Primal-Dual Convex Optimization Framework* In Advances in Neural Information Processing Systems 28, 2015.
- S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers* Foundations and Trends in Machine Learning, Vol. 3, No. 1, pp. 1–122, 2011.

Motivation

Motivation

- Evaluating the *proximal operator is costly* for many real world constrained optimization problems. This lecture covers the basics of the proximal-free numerical methods for constrained convex minimization, which use *cheaper Fenchel-type oracles* as a building block.

Swiss army knife of convex formulations

A **primal problem** prototype

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \Big\{ f(\mathbf{x}) : \mathbf{A}\mathbf{x} - \mathbf{b} \in \mathcal{K},\ \mathbf{x} \in \mathcal{X} \Big\}, \tag{1}$$

- f is a proper, closed and convex function
- $\mathcal{X}$ and $\mathcal{K}$ are nonempty, closed convex sets
- $\mathbf{A} \in \mathbb{R}^{n\times p}$ and $\mathbf{b} \in \mathbb{R}^n$ are known
- An optimal solution $\mathbf{x}^\star$ to (1) satisfies $f(\mathbf{x}^\star) = f^\star$, $\mathbf{A}\mathbf{x}^\star = \mathbf{b}$ and $\mathbf{x}^\star \in \mathcal{X}$

Swiss army knife of convex formulations

A **primal problem** prototype

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \Big\{ f(\mathbf{x}) : \mathbf{A}\mathbf{x} - \mathbf{b} \in \mathcal{K},\ \mathbf{x} \in \mathcal{X} \Big\}, \tag{1}$$

- f is a proper, closed and convex function
- $\mathcal{X}$ and $\mathcal{K}$ are nonempty, closed convex sets
- $\mathbf{A} \in \mathbb{R}^{n\times p}$ and $\mathbf{b} \in \mathbb{R}^n$ are known
- An optimal solution $\mathbf{x}^\star$ to (1) satisfies $f(\mathbf{x}^\star) = f^\star$, $\mathbf{A}\mathbf{x}^\star = \mathbf{b}$ and $\mathbf{x}^\star \in \mathcal{X}$

Recall: Definition of ϵ-accurate solutions [6]

Given a numerical tolerance $\epsilon \geq 0$, a point $\mathbf{x}^\star_\epsilon \in \mathbb{R}^p$ is called an ϵ-solution of (1) if

$$\begin{cases} f(\mathbf{x}^\star_\epsilon) - f^\star \leq \epsilon & \text{(objective residual)},\\ \mathrm{dist}\,(\mathbf{A}\mathbf{x}^\star_\epsilon - \mathbf{b}, \mathcal{K}) \leq \epsilon & \text{(feasibility gap)},\\ \mathbf{x}^\star_\epsilon \in \mathcal{X} & \text{(exact feasibility for the simple set)}. \end{cases}$$

- When $\mathbf{x}^\star$ is unique, we can also obtain $\|\mathbf{x}^\star_\epsilon - \mathbf{x}^\star\| \leq \epsilon$ (iterate residual).

- ϵ can be different for the objective, feasibility gap, or the iterate residual.

Recall the proximal operator

Proximal operator

Most primal dual methods require the computation of the prox-operator of f

$$\mathrm{prox}_f(\mathbf{x}) := \arg\min_{\mathbf{z}}\{f(\mathbf{z}) + (1/2)\|\mathbf{z}-\mathbf{x}\|^2\}.$$

Prox-operator helps us processing nonsmooth terms “efficiently”!

Problem: Not all nonsmooth functions are proximal-friendly!

Recall the proximal operator

Proximal operator

Most primal dual methods require the computation of the prox-operator of f

$$\mathrm{prox}_f(\mathbf{x}) := \arg\min_{\mathbf{z}}\{f(\mathbf{z}) + (1/2)\|\mathbf{z}-\mathbf{x}\|^2\}.$$

Prox-operator helps us processing nonsmooth terms "efficiently"!

Problem: Not all nonsmooth functions are proximal-friendly!

Example (Nuclear norm)

For $\mathbf{X} \in \mathbb{R}^{p\times p}$,

$$f(\mathbf{X}) = \|\mathbf{X}\|_\star \quad \rightarrow \quad \mathrm{prox}_f(\mathbf{X}) = \mathtt{SingValThreshold}(\mathbf{X}, 1).$$

Requires computation of the singular value decomposition! $\quad \rightarrow \quad \mathcal{O}(p^3)$

Recall the proximal operator

Proximal operator

Most primal dual methods require the computation of the prox-operator of f

$$\mathrm{prox}_f(\mathbf{x}) := \arg\min_{\mathbf{z}}\{f(\mathbf{z}) + (1/2)\|\mathbf{z}-\mathbf{x}\|^2\}.$$

Prox-operator helps us processing nonsmooth terms "efficiently"!

Problem: Not all nonsmooth functions are proximal-friendly!

Example (Nuclear norm)

For $\mathbf{X} \in \mathbb{R}^{p\times p}$,

$$f(\mathbf{X}) = \|\mathbf{X}\|_\star \quad \rightarrow \quad \mathrm{prox}_f(\mathbf{X}) = \mathtt{SingValThreshold}(\mathbf{X}, 1).$$

Requires computation of the singular value decomposition! $\quad \rightarrow \quad \mathcal{O}(p^3)$

Can we avoid the prox-operator for something cheaper?

Frank-Wolfe's method: Earliest example

Problem setting

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \Big\{ f(\mathbf{x}) : \mathbf{x} \in \mathcal{X} \Big\}, \tag{2}$$

Assumptions

- $\mathcal{X}$ is nonempty, convex, closed and bounded.
- $f \in \mathcal{F}_L^{1,1}(\mathbb{R}^p)$ (i.e., convex with Lipschitz gradient).
- Note also that $\mathbf{A}\mathbf{x} - \mathbf{b} \in \mathcal{K}$ is missing from our prototype problem.

Frank-Wolfe's method (see [3] for a review)

Conditional gradient method (CGM)

1. Choose $\mathbf{x}^0 \in \mathcal{X}$.
2. For $k = 0, 1, \ldots$ perform:

$$\begin{cases} \hat{\mathbf{x}}^k & := \arg\min_{\mathbf{x}\in\mathcal{X}} \nabla f(\mathbf{x}^k)^T \mathbf{x}, \\ \mathbf{x}^{k+1} & := (1-\gamma_k)\mathbf{x}^k + \gamma_k \hat{\mathbf{x}}^k, \end{cases}$$

where $\gamma_k := \frac{2}{k+2}$ is a given relaxation parameter.

Frank-Wolfe's method: Earliest example

Problem setting

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \Big\{ f(\mathbf{x}) : \mathbf{x} \in \mathcal{X} \Big\}, \tag{2}$$

Assumptions

- $\mathcal{X}$ is nonempty, convex, closed and bounded.
- $f \in \mathcal{F}_L^{1,1}(\mathbb{R}^p)$ (i.e., convex with Lipschitz gradient).
- Note also that $\mathbf{A}\mathbf{x} - \mathbf{b} \in \mathcal{K}$ is missing from our prototype problem.

Frank-Wolfe's method (see [3] for a review)

Conditional gradient method (CGM)
1. Choose $\mathbf{x}^0 \in \mathcal{X}$.   **2.** For $k = 0, 1, \ldots$ perform: $\begin{cases} \hat{\mathbf{x}}^k & := \arg\min_{\mathbf{x}\in\mathcal{X}} \nabla f(\mathbf{x}^k)^T \mathbf{x}, (*) \\ \mathbf{x}^{k+1} & := (1-\gamma_k)\mathbf{x}^k + \gamma_k \hat{\mathbf{x}}^k, \end{cases}$ where $\gamma_k := \frac{2}{k+2}$ is a given relaxation parameter.

When $\mathcal{X}$ is **nuclear-norm** ball, $\hat{\mathbf{x}}^k$ corresponds to **rank-1 updates!**

Recall: Fenchel conjugate

We need the definition of **Fenchel conjugation** and its basic properties to show the correspondence between CGM and DSM.

Definition

Let $\mathcal{Q}$ be a predefined Euclidean space and Q^* be its dual space. Given a proper, closed and convex function $f : \mathcal{Q} \to \mathbb{R} \cup \{+\infty\}$, the function $f^* : Q^* \to \mathbb{R} \cup \{+\infty\}$ such that

$$f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \mathrm{dom}(f)} \left\{ \mathbf{y}^T\mathbf{x} - f(\mathbf{x}) \right\}$$

is called the Fenchel conjugate (or conjugate) of f.

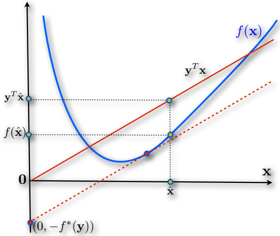


Figure: The conjugate function $f^*(\mathbf{y})$ is the maximum gap between the linear function $\mathbf{x}^T\mathbf{y}$ (red line) and $f(\mathbf{x})$.

- f^* is a convex and lower, semicontinuous function by construction (as the supremum of affine functions of $\mathbf{y}$).
- The conjugate of the conjugate of a convex function f is ... the same function f; i.e., $f^{**} = f$ for $f \in \mathcal{F}(\mathcal{Q})$.

*Basic properties of Fenchel conjugation

Property 1: Fenchel-Young inequality

Let $f : \mathcal{Q} \to \mathbb{R} \cup \{+\infty\}$ and $f^* : Q^* \to \mathbb{R} \cup \{+\infty\}$ be a function and its conjugation; here Q^* be the dual space of $\mathcal{Q}$. Then, the following inequality holds true:

$$f(\mathbf{x}) + f^*(\mathbf{y}) \geq \mathbf{x}^T\mathbf{y}, \quad \forall \mathbf{x} \in Q, \mathbf{y} \in Q^*.$$

Property 2: Subgradient property

Let $\mathbf{y} \in \partial f(\mathbf{x})$ for some $\mathbf{x} \in \mathrm{dom}(f)$. Then $\mathbf{y} \in \mathrm{dom}(f^*)$ and vise versa. Moreover, we have

$$\mathbf{u} \in \partial f(\mathbf{x}) \Leftrightarrow \mathbf{x} \in \partial f^*(\mathbf{u}).$$

Property 3: Duality of strong convexity and Lipschitz smoothness [4]

Let f be a convex and lower semi-continuos function. Then, strong convexity and Lipschitz gradients are dual in the following sense:

$$f \textit{ has Lipschitz continuos gradients} \iff f^* \textit{ is strongly convex}$$

$$f \textit{ is strongly convex} \iff f^* \textit{ has Lipschitz continuos gradients}$$

Towards Fenchel-type operators

Generalized sharp operators [8]

We define the (generalized) sharp operator of a convex function f as follows:

$$\boxed{[\mathbf{z}]^{\sharp}_{f} := \arg\min_{\mathbf{x}} \left\{ f(\mathbf{x}) - \langle \mathbf{x}, \mathbf{z} \rangle \right\}.}$$

Special case:

- [indicator function] If $f(\mathbf{x}) = \delta_{\mathcal{X}}(\mathbf{x}) \quad \rightarrow \quad [-\mathbf{x}]^{\sharp}_{f}$ is linear minimization oracle.

Towards Fenchel-type operators

Generalized sharp operators [8]

We define the (generalized) sharp operator of a convex function f as follows:

$$\boxed{[\mathbf{z}]^\sharp_f := \underset{\mathbf{x}}{\operatorname{argmin}} \left\{ f(\mathbf{x}) - \langle \mathbf{x}, \mathbf{z} \rangle \right\}.}$$

Special case:

- [indicator function] If $f(\mathbf{x}) = \delta_{\mathcal{X}}(\mathbf{x}) \quad \rightarrow \quad [-\mathbf{x}]^\sharp_f$ is linear minimization oracle.

Example (Nuclear norm)

Let σ, $\mathbf{u}$ and $\mathbf{v}$ represent the largest singular value and the associated right and left singular vectors of a matrix $\mathbf{X} \in \mathbb{R}^{p\times p}$ respectively:

$$[\mathbf{u}, \sigma, \mathbf{v}] = \mathtt{svds}(\mathbf{X}, 1)$$

- If $\phi(\mathbf{X}) = \delta_{\mathcal{X}}(\mathbf{X})$ with $\mathcal{X} := \{\mathbf{X} \in \mathbb{R}^{p\times p} : \|\mathbf{X}\|_\star \leq \kappa\}$, then $\kappa \mathbf{u}\mathbf{v}^T \in [\mathbf{X}]^\sharp_\phi$
- If $\psi(\mathbf{X}) = \frac{1}{2}\|\mathbf{X}\|^2_\star$, then $\sigma \mathbf{u}\mathbf{v}^T \in [\mathbf{X}]^\sharp_\psi$

Computation of $[\mathbf{X}]^\sharp_\phi$ and $[\mathbf{X}]^\sharp_\psi$ are essentially the same.

Revisiting Frank-Wolfe's method

Problem setting

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) : \mathbf{x} \in \mathcal{X} \right\},$$

Assumptions

- $\mathcal{X}$ is nonempty, convex, closed and bounded.
- $f \in \mathcal{F}_L^{1,1}(\mathbb{R}^p)$ (i.e., convex with Lipschitz gradient).
- Note that $\mathbf{A}\mathbf{x} - \mathbf{b} \in \mathcal{K}$ is missing from our prototype problem

Frank-Wolfe's method (see [3] for a review)

Conditional gradient method (CGM)

1. Choose $\mathbf{x}^0 \in \mathcal{X}$.

2. For $k = 0, 1, \ldots$ perform:

$$\begin{cases} \hat{\mathbf{x}}^k & := \arg\min_{\mathbf{x}\in\mathcal{X}} \nabla f(\mathbf{x}^k)^T \mathbf{x} \quad \equiv [-\nabla f(\mathbf{x}^k)]^\sharp_{\delta_\mathcal{X}}, \\ \mathbf{x}^{k+1} & := (1-\gamma_k)\mathbf{x}^k + \gamma_k \hat{\mathbf{x}}^k, \end{cases}$$

where $\gamma_k := \frac{2}{k+2}$ is a given relaxation parameter.

$$[\mathbf{z}]^\sharp_{\delta_\mathcal{X}} := \operatorname*{argmin}_{\mathbf{x}} \left\{ \delta_\mathcal{X}(\mathbf{x}) - \langle \mathbf{x}, \mathbf{z} \rangle \right\}.$$

Revisiting Frank-Wolfe's method

Problem setting

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) : \mathbf{x} \in \mathcal{X} \right\},$$

Assumptions

- $\mathcal{X}$ is nonempty, convex, closed and bounded.
- $f \in \mathcal{F}_L^{1,1}(\mathbb{R}^p)$ (i.e., convex with Lipschitz gradient).

Next: Constrained problem $\mathbf{A}\mathbf{x} - \mathbf{b} \in \mathcal{K}$ and nonsmooth $f(\mathbf{x})$ with the sharp-operator

Frank-Wolfe's method (see [3] for a review)

Conditional gradient method (CGM)

1. Choose $\mathbf{x}^0 \in \mathcal{X}$.
2. For $k = 0, 1, \ldots$ perform:

$$\begin{cases} \hat{\mathbf{x}}^k & := \arg\min_{\mathbf{x}\in\mathcal{X}} \nabla f(\mathbf{x}^k)^T\mathbf{x} \quad \equiv [-\nabla f(\mathbf{x}^k)]^\sharp_{\delta_{\mathcal{X}}}, \\ \mathbf{x}^{k+1} & := (1-\gamma_k)\mathbf{x}^k + \gamma_k\hat{\mathbf{x}}^k, \end{cases}$$

where $\gamma_k := \frac{2}{k+2}$ is a given relaxation parameter.

$$[\mathbf{z}]^\sharp_{\delta_{\mathcal{X}}} := \operatorname*{argmin}_{\mathbf{x}} \left\{ \delta_{\mathcal{X}}(\mathbf{x}) - \langle \mathbf{x}, \mathbf{z} \rangle \right\}.$$

*CGM is dual averaging subgradient method

$$\min_{\mathbf{r},\mathbf{x}} \{f(\mathbf{x}) : \mathbf{x} = \mathbf{r}, \mathbf{r} \in \mathcal{X}\}$$

Dual averaging subgradient method:

For $k = 0$ **to** $k_{\max}$:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \gamma_k \nabla d(\boldsymbol{\lambda}^k)$$

$$\boldsymbol{\lambda}^{k+1} = \arg\max_{\boldsymbol{\lambda}} \left\{ \langle \boldsymbol{\lambda}, \mathbf{x}^{k+1} \rangle - \beta_k \phi(\boldsymbol{\lambda}) \right\}$$

End for

$\mathbf{x}^0 = 0$, $\beta_{k+1} \leq \beta_k$, and ϕ is a strongly convex function (that we can choose).

*CGM is dual averaging subgradient method

$$\min_{\mathbf{r},\mathbf{x}} \{ f(\mathbf{x}) : \mathbf{x} = \mathbf{r}, \mathbf{r} \in \mathcal{X} \}$$

Dual averaging subgradient method:

For $k = 0$ **to** $k_{\max}$:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \gamma_k \nabla d(\boldsymbol{\lambda}^k)$$

$$\boldsymbol{\lambda}^{k+1} = \arg\max_{\boldsymbol{\lambda}} \left\{ \langle \boldsymbol{\lambda}, \mathbf{x}^{k+1} \rangle - f^*(\boldsymbol{\lambda}) \right\}$$

End for

Choose

$\beta_k = 1$,

$\phi = f^*$ (strongly convex due to Fenchel duality, since f is smooth)

*CGM is dual averaging subgradient method

$$\min_{\mathbf{r},\mathbf{x}} \{f(\mathbf{x}) : \mathbf{x} = \mathbf{r}, \mathbf{r} \in \mathcal{X}\}$$

Dual averaging subgradient method:

For $k = 0$ **to** $k_{\max}$:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \gamma_k(\mathbf{x}^*(\boldsymbol{\lambda}^k) - \mathbf{r}^*(\boldsymbol{\lambda}^k))$$

$$\boldsymbol{\lambda}^{k+1} = \arg\max_{\boldsymbol{\lambda}} \left\{\langle \boldsymbol{\lambda}, \mathbf{x}^{k+1}\rangle - f^*(\boldsymbol{\lambda})\right\}$$

End for

- Augment the dual:

$$d(\boldsymbol{\lambda}) = \min_{\mathbf{r}} \underbrace{\{f(\mathbf{r}) - \langle \boldsymbol{\lambda}, \mathbf{r}\rangle\}}_{-f^*(\boldsymbol{\lambda})} + \min_{\mathbf{x}} \{\langle \boldsymbol{\lambda}, \mathbf{x}\rangle : \ \mathbf{x} \in \mathcal{X}\}$$

$$\nabla d(\boldsymbol{\lambda}^k) = \mathbf{x}^*(\boldsymbol{\lambda}^k) - \mathbf{r}^*(\boldsymbol{\lambda}^k)$$

$$\boldsymbol{\lambda}^k = \nabla f(\mathbf{r}^*(\boldsymbol{\lambda}^k)) \quad \Longleftrightarrow \quad \mathbf{r}^*(\boldsymbol{\lambda}^k) \in \partial f^*(\boldsymbol{\lambda}^k)$$

Due to Fenchel duality.

*CGM is dual averaging subgradient method

$$\min_{\mathbf{r},\mathbf{x}} \{f(\mathbf{x}) : \mathbf{x} = \mathbf{r}, \mathbf{r} \in \mathcal{X}\}$$

Dual averaging subgradient method:

For $k = 0$ **to** $k_{\max}$:

$\quad \mathbf{x}^{k+1} = \mathbf{x}^k + \gamma_k(\mathbf{x}^*(\boldsymbol{\lambda}^k) - \mathbf{r}^*(\boldsymbol{\lambda}^k))$

$\quad \boldsymbol{\lambda}^{k+1} = \nabla f(\mathbf{x}^{k+1})$

End for

$$\boldsymbol{\lambda}^{k+1} = \arg\max_{\boldsymbol{\lambda}} \left\{\langle \boldsymbol{\lambda}, \mathbf{x}^{k+1}\rangle - f^*(\boldsymbol{\lambda})\right\}$$

$$\mathbf{x}^{k+1} \in \partial f^*(\boldsymbol{\lambda}^{k+1}) \quad \Longleftrightarrow \quad \boldsymbol{\lambda}^{k+1} = \nabla f(\mathbf{x}^{k+1})$$

Due to Fenchel duality.

*CGM is dual averaging subgradient method

$$\min_{\mathbf{r},\mathbf{x}} \{f(\mathbf{x}) : \mathbf{x} = \mathbf{r}, \mathbf{r} \in \mathcal{X}\}$$

Dual averaging subgradient method: $\Longrightarrow$ CGM

For $k = 0$ **to** $k_{\max}$:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \gamma_k(\mathbf{x}^*(\boldsymbol{\lambda}^k) - \mathbf{x}^k)$$
$$\boldsymbol{\lambda}^{k+1} = \nabla f(\mathbf{x}^{k+1})$$

End for

$$\boldsymbol{\lambda}^{k+1} = \arg\max_{\boldsymbol{\lambda}} \left\{ \langle \boldsymbol{\lambda}, \mathbf{x}^{k+1} \rangle - f^*(\boldsymbol{\lambda}) \right\}$$

$$\mathbf{x}^{k+1} \in \partial f^*(\boldsymbol{\lambda}^{k+1}) \quad \Longleftrightarrow \quad \boldsymbol{\lambda}^{k+1} = \nabla f(\mathbf{x}^{k+1})$$

Due to Fenchel duality.

We can choose $\mathbf{r}^*(\boldsymbol{\lambda}^k) = \mathbf{x}^k$ since $\mathbf{r}^*(\boldsymbol{\lambda}^k) \in \partial f^*(\boldsymbol{\lambda}^k)$

Finding an optimal solution

A plausible algorithmic strategy for $\min_{\mathbf{x}\in\mathcal{X}}\{f(\mathbf{x}) : \mathbf{A}\mathbf{x} = \mathbf{b}\}$:

A natural minimax formulation:

$$(\mathbf{x}^\star, \lambda^\star) \in \arg\max_{\lambda} \min_{\mathbf{x}\in\mathcal{X}} \{\mathcal{L}(\mathbf{x}, \lambda) := f(\mathbf{x}) + \langle \lambda, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle\}.$$

Lagrangian subproblem: $\mathbf{x}^*(\lambda) \in \arg\min_{\mathbf{x}\in\mathcal{X}} \mathcal{L}(\mathbf{x}, \lambda)$

Dual problem: $\lambda^\star \in \arg\max_{\lambda} \{d(\lambda) := \mathcal{L}(\mathbf{x}^*(\lambda), \lambda)\}$

- λ is called the Lagrange multiplier.
- The function $d(\lambda)$ is called the dual function, and it is concave!
- The optimal dual objective value is $d^\star = d(\lambda^\star)$.

Our strategy $\Rightarrow$ Make progress on the dual and obtain the primal solution

For notational simplicity, we denote $g(\boldsymbol{\lambda}) = -d(\boldsymbol{\lambda})$ and consider convex minimization.

Finding an optimal solution

A plausible algorithmic strategy for $\min_{\mathbf{x}\in\mathcal{X}}\{f(\mathbf{x}) : \mathbf{A}\mathbf{x} = \mathbf{b}\}$:

A natural minimax formulation:

$$(\mathbf{x}^\star, \lambda^\star) \in \arg\max_{\lambda} \min_{\mathbf{x}\in\mathcal{X}} \{\mathcal{L}(\mathbf{x}, \lambda) := f(\mathbf{x}) + \langle \lambda, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle\}.$$

Lagrangian subproblem: $\mathbf{x}^*(\lambda) \in \arg\min_{\mathbf{x}\in\mathcal{X}} \mathcal{L}(\mathbf{x}, \lambda)$

Dual problem: $\lambda^\star \in \arg\max_{\lambda} \{d(\lambda) := \mathcal{L}(\mathbf{x}^*(\lambda), \lambda)\}$

- λ is called the Lagrange multiplier.
- The function $d(\lambda)$ is called the dual function, and it is concave!
- The optimal dual objective value is $d^\star = d(\lambda^\star)$.

Our strategy $\Rightarrow$ Make progress on the dual and obtain the primal solution

For notational simplicity, we denote $g(\boldsymbol{\lambda}) = -d(\boldsymbol{\lambda})$ and consider convex minimization.

Challenges for the plausible strategy above

1. Establishing its correctness
2. Computational efficiency of finding an $\bar{\epsilon}$-approximate optimal dual solution $\lambda^\star_{\bar{\epsilon}}$
3. Mapping $\lambda^\star_{\bar{\epsilon}} \to \mathbf{x}^\star_{\bar{\epsilon}}$

Finding an optimal solution

A plausible algorithmic strategy for $\min_{\mathbf{x}\in\mathcal{X}}\{f(\mathbf{x}) : \mathbf{A}\mathbf{x} = \mathbf{b}\}$:

A natural minimax formulation:

$$(\mathbf{x}^\star, \lambda^\star) \in \arg\max_{\lambda} \min_{\mathbf{x}\in\mathcal{X}}\{\mathcal{L}(\mathbf{x}, \lambda) := f(\mathbf{x}) + \langle \lambda, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle\}.$$

Lagrangian subproblem: $\mathbf{x}^*(\lambda) \in \arg\min_{\mathbf{x}\in\mathcal{X}} \mathcal{L}(\mathbf{x}, \lambda)$

Dual problem: $\lambda^\star \in \arg\max_\lambda \{d(\lambda) := \mathcal{L}(\mathbf{x}^*(\lambda), \lambda)\}$

- λ is called the Lagrange multiplier.
- The function $d(\lambda)$ is called the dual function, and it is concave!
- The optimal dual objective value is $d^\star = d(\lambda^\star)$.

Our strategy $\Rightarrow$ Make progress on the dual and obtain the primal solution

For notational simplicity, we denote $g(\boldsymbol{\lambda}) = -d(\boldsymbol{\lambda})$ and consider convex minimization.

Challenges for the plausible strategy above

1. Establishing its correctness: Assume $f^\star > -\infty$ and Slater's condition for $f^\star = d^\star$
2. Computational efficiency of finding an $\bar{\epsilon}$-approximate optimal dual solution $\lambda^\star_{\bar{\epsilon}}$
3. Mapping $\lambda^\star_{\bar{\epsilon}} \to \mathbf{x}^\star_{\bar{\epsilon}}$

Efficiency considerations for the dual problem

If $g(\boldsymbol{\lambda})$ is non-smooth (with bounded subgradients)

$$\exists G > 0 : \quad \|\mathbf{v}\|_2 \leq G, \qquad \forall \mathbf{v} \in \partial g(\boldsymbol{\lambda}), \ \forall \boldsymbol{\lambda} \in \mathbb{R}^n.$$

- Subgradient method in the dual $\quad \rightarrow \quad \mathcal{O}\left(\frac{1}{\epsilon^2}\right)$

Efficiency considerations for the dual problem

If $g(\boldsymbol{\lambda})$ is non-smooth (with bounded subgradients)

$$\exists G > 0: \quad \|\mathbf{v}\|_2 \leq G, \quad \forall \mathbf{v} \in \partial g(\boldsymbol{\lambda}),\ \forall \boldsymbol{\lambda} \in \mathbb{R}^n.$$

- Subgradient method in the dual $\quad \rightarrow \quad \mathcal{O}\left(\frac{1}{\epsilon^2}\right)$

If $g(\boldsymbol{\lambda})$ is smooth (Lipschitz gradients)

$$\|\nabla g(\boldsymbol{\lambda}) - \nabla g(\boldsymbol{\eta})\|_2 \leq L\|\boldsymbol{\lambda} - \boldsymbol{\eta}\|_2, \quad \forall \boldsymbol{\lambda}, \boldsymbol{\eta} \in \mathbb{R}^n.$$

- Accelerated gradient method in the dual $\quad \rightarrow \quad \mathcal{O}\left(\frac{1}{\sqrt{\epsilon}}\right)$.

Efficiency considerations for the dual problem

If $g(\boldsymbol{\lambda})$ is non-smooth (with bounded subgradients)

$$\exists G > 0 : \quad \|\mathbf{v}\|_2 \le G, \quad \forall \mathbf{v} \in \partial g(\boldsymbol{\lambda}), \ \forall \boldsymbol{\lambda} \in \mathbb{R}^n.$$

- Subgradient method in the dual $\quad \rightarrow \quad \mathcal{O}\left(\frac{1}{\epsilon^2}\right)$

Our strategy: Hölder smoothness in the dual

We assume that $\nabla g(\boldsymbol{\lambda})$ is Hölder continuous for some $\nu \in [0, 1]$:

$$\|\nabla g(\boldsymbol{\lambda}) - \nabla g(\boldsymbol{\eta})\|_2 \le M_\nu \|\boldsymbol{\lambda} - \boldsymbol{\eta}\|_2^\nu, \quad \forall \boldsymbol{\lambda}, \boldsymbol{\eta} \in \mathbb{R}^n$$

- Theoretical lowerbound: $\mathcal{O}\left(\left(\frac{1}{\epsilon}\right)^{\frac{2}{1+3\nu}}\right)$.
  - $\nu = 0$ is equivalent to the bounded (sub)gradient assumption.
  - $\nu = 1$ is equivalent to the Lipschitz gradients assumption.

If $g(\boldsymbol{\lambda})$ is smooth (Lipschitz gradients)

$$\|\nabla g(\boldsymbol{\lambda}) - \nabla g(\boldsymbol{\eta})\|_2 \le L \|\boldsymbol{\lambda} - \boldsymbol{\eta}\|_2, \quad \forall \boldsymbol{\lambda}, \boldsymbol{\eta} \in \mathbb{R}^n.$$

- Accelerated gradient method in the dual $\quad \rightarrow \quad \mathcal{O}\left(\frac{1}{\sqrt{\epsilon}}\right)$.

Brief detour: Exploring the smoothness in depth

Consider the following unconstrained convex minimization

$$\min_{\mathbf{x}\in\mathbb{R}^p} g(\mathbf{x})$$

Practical difficulty of using Hölder continuity

Hölder continuous (sub)gradients ensures the following basic surrogate for any $\mathbf{x}, \mathbf{y} \in \mathcal{X}$:

$$g(\mathbf{y}) \leq g(\mathbf{x}) + \langle \nabla g(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{M_\nu}{1+\nu} \|\mathbf{x} - \mathbf{y}\|^{1+\nu} \qquad (3)$$

In practice, smoothness parameters ν and M_ν are usually not known.

Brief detour: Exploring the smoothness in depth

Consider the following unconstrained convex minimization

$$\min_{\mathbf{x}\in\mathbb{R}^p} g(\mathbf{x})$$

Practical difficulty of using Hölder continuity

Hölder continuous (sub)gradients ensures the following basic surrogate for any $\mathbf{x}, \mathbf{y} \in \mathcal{X}$:

$$g(\mathbf{y}) \leq g(\mathbf{x}) + \langle \nabla g(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{M_\nu}{1+\nu} \|\mathbf{x} - \mathbf{y}\|^{1+\nu} \tag{3}$$

In practice, smoothness parameters ν and M_ν are usually not known.

Nesterov's universal gradient lemma [5].

Let g satisfy (3). Then for any $\epsilon > 0$ and

$$M \geq \left[\frac{1-\nu}{1+\nu} \cdot \frac{1}{\delta}\right]^{\frac{1-\nu}{1+\nu}} M_\nu^{\frac{2}{1+\nu}}$$

we have

$$g(\mathbf{y}) \leq g(\mathbf{x}) + \langle \nabla g(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{M}{2} \|\mathbf{x} - \mathbf{y}\|^2 + \frac{\epsilon}{2}$$

Brief detour: Exploring the smoothness in depth

Consider the following unconstrained convex minimization

$$\min_{\mathbf{x}\in\mathbb{R}^p} g(\mathbf{x})$$

Practical difficulty of using Hölder continuity

Hölder continuous (sub)gradients ensures the following basic surrogate for any $\mathbf{x}, \mathbf{y} \in \mathcal{X}$:

$$g(\mathbf{y}) \leq g(\mathbf{x}) + \langle \nabla g(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{M_\nu}{1+\nu}\|\mathbf{x} - \mathbf{y}\|^{1+\nu} \tag{3}$$

In practice, smoothness parameters ν and M_ν are usually not known.

Nesterov's universal gradient lemma [5].

Let g satisfy (3). Then for any $\epsilon > 0$ and

$$M \geq \left[\frac{1-\nu}{1+\nu} \cdot \frac{1}{\delta}\right]^{\frac{1-\nu}{1+\nu}} M_\nu^{\frac{2}{1+\nu}}$$

we have

$$\boxed{g(\mathbf{y}) \leq g(\mathbf{x}) + \langle \nabla g(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{M}{2}\|\mathbf{x} - \mathbf{y}\|^2 + \frac{\epsilon}{2}}$$

This lemma provides us the linesearch condition!

Nesterov's universal gradient methods

Universal primal gradient method (PGM)[1]
1. Choose $\mathbf{x}^0 \in \mathcal{X}$, $M_{-1} > 0$ and accuracy $\epsilon > 0$. **2.** For $k = 0, 1, \ldots$ perform: $$\mathbf{x}^{k+1} = \mathbf{x}^k - M_k^{-1}\nabla g(\mathbf{x}^k)$$ using line-search to find $M_k \geq 0.5 M_{k-1}$ that satisfies: $$g(\mathbf{x}^{k+1}) \leq g(\mathbf{x}^k) + \langle \nabla g(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^k \rangle + \frac{M_k}{2}\|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 + \frac{\epsilon}{2}$$

Nesterov's universal gradient method [5]

- Adapt to the unknown ν via an line-search strategy
- Universal since they ensure the best possible rate of convergence for each ν

[1] PGM in [5] uses the Bregman / prox setup.

Nesterov's universal gradient methods

Universal primal gradient method (PGM)[1]

1. Choose $\mathbf{x}^0 \in \mathcal{X}$, $M_{-1} > 0$ and accuracy $\epsilon > 0$.
2. For $k = 0, 1, \ldots$ perform:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - M_k^{-1}\nabla g(\mathbf{x}^k)$$

using line-search to find $M_k \geq 0.5M_{k-1}$ that satisfies:

$$g(\mathbf{x}^{k+1}) \leq g(\mathbf{x}^k) + \langle \nabla g(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^k \rangle + \frac{M_k}{2}\|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 + \frac{\epsilon}{2}$$

Nesterov's universal gradient method [5]

- Adapt to the unknown ν via an line-search strategy
- Universal since they ensure the best possible rate of convergence for each ν

Yes, there is an accelerated version [5].

[1] PGM in [5] uses the Bregman / prox setup.

Nesterov's universal gradient methods

Universal primal gradient method (PGM)[1]

1. Choose $\mathbf{x}^0 \in \mathcal{X}$, $M_{-1} > 0$ and accuracy $\epsilon > 0$.
2. For $k = 0, 1, \ldots$ perform:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - M_k^{-1} \nabla g(\mathbf{x}^k)$$

using line-search to find $M_k \geq 0.5 M_{k-1}$ that satisfies:

$$g(\mathbf{x}^{k+1}) \leq g(\mathbf{x}^k) + \langle \nabla g(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^k \rangle + \frac{M_k}{2} \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 + \frac{\epsilon}{2}$$

Nesterov's universal gradient method [5]

- Adapt to the unknown ν via an line-search strategy
- Universal since they ensure the best possible rate of convergence for each ν

Yes, there is an accelerated version [5].

New: Our FISTA variant.

[1] PGM in [5] uses the Bregman / prox setup.

Our universal primal-dual gradient methods: The main steps

$$\boxed{[\mathbf{z}]^\sharp_f := \operatorname*{argmin}_{\mathbf{x}} \{f(\mathbf{x}) - \langle \mathbf{x}, \mathbf{z}\rangle\}}$$

Universal primal-dual gradient method (UniPDGrad)

Input initial dual point $\boldsymbol{\lambda}^0$ and desired accuracy ϵ. Then, at each iteration:

1. Solve Lagrangian subproblem (i.e., evaluate the sharp operator)

$$\boxed{\mathbf{x}^*(\boldsymbol{\lambda}^k) \in \arg\min_{\mathbf{x}\in\mathcal{X}} \left\{ f(\mathbf{x}) + \langle \boldsymbol{\lambda}^k, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle \right\} \equiv \left[-\mathbf{A}^T\boldsymbol{\lambda}^k\right]^\sharp_{f+\delta_{\mathcal{X}}}}$$

2. Take a gradient step in the dual (find M_k by the inexact line-search condition)

$$\boldsymbol{\lambda}^{k+1} := \boldsymbol{\lambda}^k - \frac{1}{M_k}\nabla g(\boldsymbol{\lambda}^k) = \boldsymbol{\lambda}^k + \frac{1}{M_k}\left(\mathbf{A}\mathbf{x}^*(\boldsymbol{\lambda}^k) - \mathbf{b}\right)$$

3. Take the weighted average for primal reconstruction

$$\bar{\mathbf{x}}^k := \left(\sum_{i=0}^{k} \frac{1}{M_i}\right)^{-1} \sum_{i=0}^{k} \frac{1}{M_i}\mathbf{x}^*(\boldsymbol{\lambda}^i)$$

Summary of the algorithms and convergence guarantees - I

Universal primal-dual gradient method (UniPDGrad)
Initialization: Choose $\lambda^0 \in \mathbb{R}^n$ and $\epsilon > 0$. Estimate a value $M_{-1} < 2M_\epsilon$.
Iteration: For $k = 0, 1, \ldots$ perform:
1. *Primal step:* $\mathbf{x}^*(\boldsymbol{\lambda}^k) = [-\mathbf{A}^T\boldsymbol{\lambda}^k]^\sharp_f$
2. *Dual gradient:* $\nabla g(\boldsymbol{\lambda}^k) = \mathbf{b} - \mathbf{A}^T\mathbf{x}^*(\boldsymbol{\lambda}^k)$
3. Line-search: Find $M_k \in [0.5M_{k-1}, 2M_\epsilon]$ from line-search condition and: $\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k - M_k^{-1}\nabla g(\boldsymbol{\lambda}^k)$
4. *Primal averaging:* $\mathbf{x}^k := S_k^{-1}\sum_{j=0}^k M_j^{-1}\mathbf{x}^*(\boldsymbol{\lambda}^j)$ where $S_k = \sum_{j=0}^k M_j^{-1}$.

$$\boxed{g(\boldsymbol{\lambda}^{k+1}) \le g(\boldsymbol{\lambda}^k) + \langle \nabla g(\boldsymbol{\lambda}^k), \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\rangle + \frac{M}{2}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \frac{\epsilon}{2}}$$

Theorem [8]

$\mathbf{x}^k$ obtained by **UniPDGrad** satisfy:

$$\begin{cases} -\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|\|\boldsymbol{\lambda}^\star\| \le & f(\mathbf{x}^k) - f^\star & \le \frac{M_\epsilon\|\boldsymbol{\lambda}^0\|^2}{k+1} + \frac{\epsilon}{2}, \\ & \|\mathbf{A}\mathbf{x}^k - \mathbf{b}\| & \le \frac{4M_\epsilon\|\boldsymbol{\lambda}^0 - \boldsymbol{\lambda}^\star\|}{k+1} + \sqrt{\frac{2M_\epsilon\epsilon}{k+1}}. \end{cases}$$

Summary of the algorithms and convergence guarantees - II

Accelerated universal primal-dual gradient method (AccUniPDGrad)

Initialization: Choose $\lambda^0 \in \mathbb{R}^n$, $\epsilon > 0$. Set $t_0 = 1$. Estimate a value $M_{-1} < 2M_\epsilon$.

Iteration: For $k = 0, 1, \ldots$ perform:

1. *Primal step:* $\mathbf{x}^*(\hat{\boldsymbol{\lambda}}^k) = [-\mathbf{A}^T\hat{\boldsymbol{\lambda}}^k]^\sharp_f$,
2. *Dual gradient:* $\nabla g(\hat{\boldsymbol{\lambda}}^k) = \mathbf{b} - \mathbf{A}^T\mathbf{x}^*(\hat{\boldsymbol{\lambda}}^k)$,
3. *Line-search:* Find $M_k \in [M_{k-1}, 2M_\epsilon]$ from line-search condition and:
$$\boldsymbol{\lambda}^{k+1} = \hat{\boldsymbol{\lambda}}^k - M_k^{-1}\nabla g(\hat{\boldsymbol{\lambda}}^k),$$
4. $t_{k+1} = 0.5[1 + \sqrt{1 + 4t_k^2}]$,
5. $\hat{\boldsymbol{\lambda}}_{k+1} = \boldsymbol{\lambda}_{k+1} + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{\lambda}_{k+1} - \lambda_k)$,
6. *Primal averaging:* $\mathbf{x}^k := S_k^{-1}\sum_{j=0}^k t_j M_j^{-1}\mathbf{x}^*(\boldsymbol{\lambda}^j)$ where $S_k = \sum_{j=0}^k t_j M_j^{-1}$.

$$\boxed{g(\boldsymbol{\lambda}^{k+1}) \le g(\hat{\boldsymbol{\lambda}}^k) + \langle \nabla g(\hat{\boldsymbol{\lambda}}^k), \boldsymbol{\lambda}^{k+1} - \hat{\boldsymbol{\lambda}}^k\rangle + \frac{M}{2}\|\boldsymbol{\lambda}^{k+1} - \hat{\boldsymbol{\lambda}}^k\|^2 + \frac{\epsilon}{2t_k}}$$

Theorem [8]

$\mathbf{x}^k$ obtained by **AccUniProx** satisfy:

$$\begin{cases} -\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|\|\boldsymbol{\lambda}^\star\| \le & f(\mathbf{x}^k) - f^\star & \le \frac{4M_\epsilon\|\boldsymbol{\lambda}^0\|^2}{(k+2)^{\frac{1+3\nu}{1+\nu}}} + \frac{\epsilon}{2}, \\ & \|\mathbf{A}\mathbf{x}^k - \mathbf{b}\| & \le \frac{16M_\epsilon\|\lambda^0 - \lambda^\star\|}{(k+2)^{\frac{1+3\nu}{1+\nu}}} + \sqrt{\frac{8M_\epsilon\epsilon}{(k+2)^{\frac{1+3\nu}{1+\nu}}}}. \end{cases}$$

The general constraint case

Handling to the constraint $\mathbf{A}\mathbf{x} - \mathbf{b} \in \mathcal{K}$

the universal dual accelerated gradient method:

$$\begin{cases} t_k & := 0.5\left(1 + \sqrt{1 + 4t_{k-1}^2}\right) \\ \hat{\lambda}^k & := \bar{\lambda}^k + \frac{t_{k-1}-1}{t_k}\left(\bar{\lambda}^k - \hat{\lambda}^{k-1}\right) \\ \lambda^{k+1} & := \hat{\lambda}^k + \frac{1}{M_k}\Big(\mathbf{A}\mathbf{x}^*(\hat{\lambda}^k) - \mathbf{b}\Big). \end{cases}$$

The general constraint case

Handling to the constraint $\mathbf{A}\mathbf{x} - \mathbf{b} \in \mathcal{K}$

Only one prox change in the universal dual accelerated gradient method:

$$\begin{cases} t_k & := 0.5\left(1 + \sqrt{1 + 4t_{k-1}^2}\right) \\ \hat{\lambda}^k & := \bar{\lambda}^k + \frac{t_{k-1}-1}{t_k}\left(\bar{\lambda}^k - \hat{\lambda}^{k-1}\right) \\ \lambda^{k+1} & := \mathrm{prox}_{M_k^{-1}h}\left(\hat{\lambda}^k + \frac{1}{M_k}\left(\mathbf{A}\mathbf{x}^*(\hat{\lambda}^k) - \mathbf{b}\right)\right). \end{cases}$$

Here, h is defined by $h(\lambda) := \sup_{\mathbf{r} \in \mathcal{K}} \langle \lambda, \mathbf{r} \rangle$.

Theoretical guarantees

Universality of the method [8]

We derive the following worst-case iteration complexity results to obtain $\epsilon-$accurate solution $\mathbf{x}^k$ in the sense

$$|f(\mathbf{x}^k) - f^\star| \leq \epsilon, \quad \mathtt{dist}(\mathbf{A}\mathbf{x}^k - \mathbf{b}, \mathcal{K}) \leq \epsilon \quad \text{and} \quad \mathbf{x}^k \in \mathcal{X}$$

$$\begin{cases} \textbf{UniPDGrad:} & \mathcal{O}\left(D^2_{\Lambda^\star} \inf_{0\leq\nu\leq 1} \left(\frac{M_\nu}{\epsilon}\right)^{\frac{2}{1+\nu}}\right), & \text{optimal for } \nu = 0 \\ \textbf{AccUniPDGrad:} & \mathcal{O}\left((2D_{\Lambda^\star})^{\frac{2+2\nu}{1+3\nu}} \inf_{0\leq\nu\leq 1} \left(\frac{M_\nu}{\epsilon}\right)^{\frac{2}{1+3\nu}}\right), & \text{optimal for } \nu \in [0, 1] \end{cases}$$

where $D_{\Lambda^\star} := \frac{4\sqrt{2}\|\boldsymbol{\lambda}^\star\|}{-1+\sqrt{1+8\frac{\|\boldsymbol{\lambda}^\star\|}{\max\{\|\boldsymbol{\lambda}^\star\|,1\}}}}$.

Note:
- Both UniPDGrad and AccUniPDGrad require 2 sharp operators queries per iteration on average.

*Example: Phase retrieval

Phase retrieval

Aim: Recover signal $\mathbf{x}^\natural \in \mathbb{C}^p$ from the measurements $\mathbf{b} \in \mathbb{R}^n$:

$$b_i = \left|\langle \mathbf{a}_i, \mathbf{x}^\natural \rangle\right|^2 + \omega_i.$$

($\mathbf{a}_i \in \mathbb{C}^p$ are known measurement vectors, ω_i models noise).
• Non-linear measurements $\rightarrow$ **non-convex** maximum likelihood estimators.

PhaseLift [1]

Phase retrieval can be solved as a convex matrix completion problem, following a combination of

- semidefinite relaxation $(\mathbf{x}^\natural {\mathbf{x}^\natural}^H = \mathbf{X}^\natural)$
- convex relaxation $(\texttt{rank} \rightarrow \|\cdot\|_*)$

albeit in terms of the lifted variable $\mathbf{X} \in \mathbb{C}^{p\times p}$.

Example: Phase retrieval - II

Problem formulation

We solve the following PhaseLift variant:

$$f^\star := \min_{\mathbf{X} \in \mathbb{C}^{p \times p}} \left\{ \frac{1}{2} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|_2^2 : \quad \|\mathbf{X}\|_* \leq \kappa, \quad \mathbf{X} \succeq 0 \right\}. \tag{4}$$

Experimental setup [7]

Coded diffraction pattern measurements, $\mathbf{b} = [\mathbf{b}_1, \ldots, \mathbf{b}_L]$ with $L = 20$ different masks

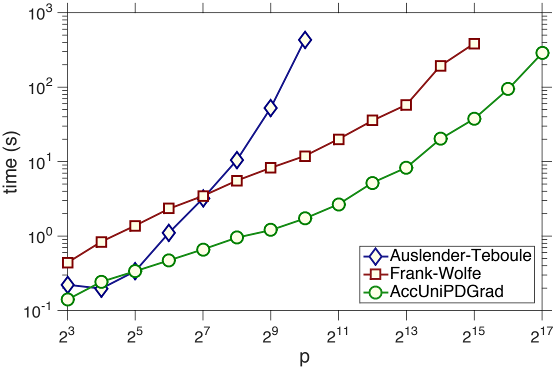
$$\mathbf{b}_\ell = |\mathtt{fft}(\mathbf{d}_\ell^H \odot \mathbf{x}^\natural)|^2$$

$\rightarrow$ $\odot$ denotes Hadamard product; $|\cdot|^2$ applies element-wise

$\rightarrow$ $\mathbf{d}_\ell$ are randomly generated octonary masks (distributions as proposed in [1])

$\rightarrow$ Parametric choices: $\lambda^0 = \mathbf{0}^n$; $\quad \epsilon = 10^{-2}$; $\quad \kappa = \mathtt{mean}(\mathbf{b})$.

Example: Phase retrieval - III



Test with synthetic data: Prox vs sharp

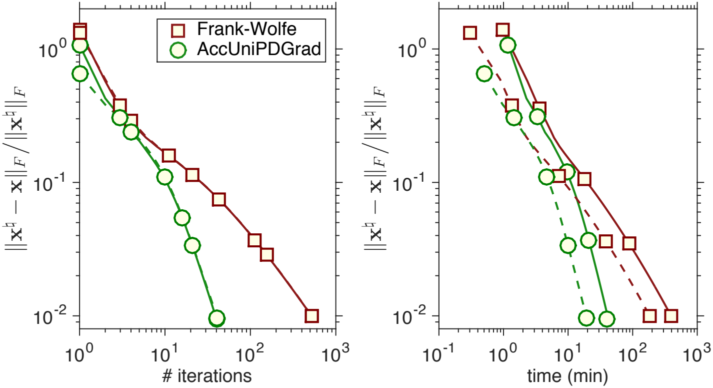
$\rightarrow$ Synthetic data: $\mathbf{x}^\natural = \mathtt{randn}(p, 1) + i \cdot \mathtt{randn}(p, 1)$.

$\rightarrow$ Stopping criteria: $\frac{\|\mathbf{x}^\natural - \mathbf{x}^k\|_2}{\|\mathbf{x}^\natural\|_2} \leq 10^{-2}$.

$\rightarrow$ Averaged over 10 Monte-Carlo iterations.

Note that the problem is $p \times p$ dimensional!

Scalability example: Phase retrieval - IV



Test with images

We use real images of

- EPFL campus of size 1280×720 $\rightarrow$ $p^2 \approx 10^{12}$ (dashed lines)
- Milky Way galaxy of size 1920×1080 $\rightarrow$ $p^2 \approx 4 \cdot 10^{12}$ (solid lines)

Example: Phase retrieval - V

EPFL campus image of size 1280×720, reconstructed in 20 minutes by 41 iterations of AccUniPDGrad: $\mathtt{PSNR} = 45.54$ dB

Scalability example: Phase retrieval - VI

Milky Way galaxy image of size 1920×1080, reconstructed in 42 minutes by 40 iterations of AccUniPDGrad: $\mathtt{PSNR} = 54.44$ dB

Example: Quantum tomography with Pauli operators - I

Problem formulation

Let $\mathbf{X}^\natural \in \mathcal{S}_+^p$ be a density matrix which characterizes a $q-$qubit quantum system, where $p = 2^q$. Using Pauli operators $\mathcal{A}$ [2], we can deduce the state from $\mathbf{b} = \mathcal{A}(\mathbf{X}) \in \mathcal{C}^n$ based on the following convex optimization formulation:

$$\varphi^\star := \min_{\mathbf{X} \in \mathcal{S}_+^p} \left\{ \frac{1}{2} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|_2^2 : \mathrm{tr}(\mathbf{X}) = 1 \right\}. \quad (5)$$

The recovery is also robust to noise.

Example: Quantum tomography with Pauli operators - I

Problem formulation

Let $\mathbf{X}^\natural \in \mathcal{S}_+^p$ be a density matrix which characterizes a $q-$qubit quantum system, where $p = 2^q$. Using Pauli operators $\mathcal{A}$ [2], we can deduce the state from $\mathbf{b} = \mathcal{A}(\mathbf{X}) \in \mathcal{C}^n$ based on the following convex optimization formulation:

$$\varphi^\star := \min_{\mathbf{X} \in \mathcal{S}_+^p} \left\{ \frac{1}{2} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|_2^2 : \mathrm{tr}(\mathbf{X}) = 1 \right\}. \quad (5)$$

The recovery is also robust to noise.
Perfect scalability test: tuning free constraint + Lipschitz continuous gradient

Setup

Synthetic random pure quantum state (e.g., rank-1 $\mathbf{X}^\natural$) with:

- $q = 14$ qubits, that corresponds to $2^{28} = 268'435'456$ dimensional problem.
- $n := 2p\log(p) = 138'099$ number of Pauli measurements.
- Input parameters $\lambda^0 = \mathbf{0}^n$ and $\epsilon = 2 \cdot 10^{-4}$.

Example: Quantum tomography with Pauli operators - II

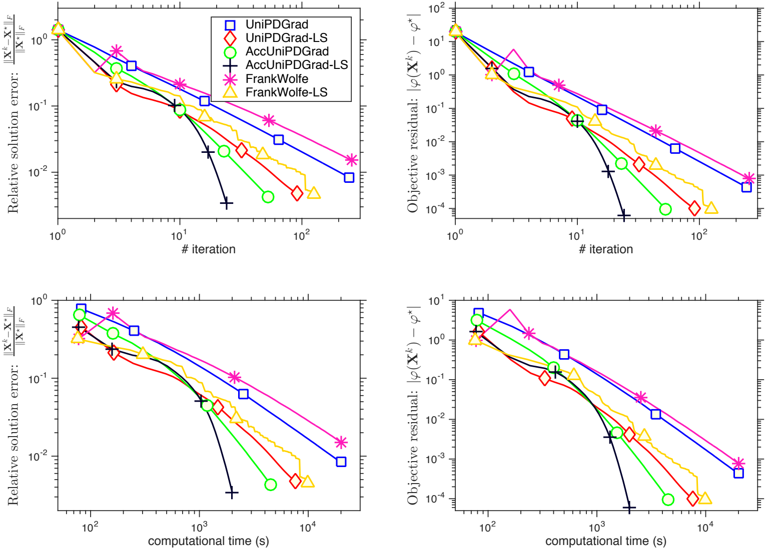


Figure: The performance of (Acc)UniPDGrad and Frank-Wolfe algorithms for (5).

Outline

Yet another template from source separation

Bonus: ADMM[2]

Primal problem with a specific decomposition structure

$$f^\star := \min_{\mathbf{x}:=(\mathbf{u},\mathbf{v})} \{ f(\mathbf{x}) := g(\mathbf{u}) + h(\mathbf{v}) : \mathbf{B}\mathbf{u} + \mathbf{C}\mathbf{v} = \mathbf{b},\ \mathbf{u} \in \mathcal{U},\ \mathbf{v} \in \mathcal{V} \}$$

- $\mathcal{X} := \mathcal{U} \times \mathcal{V}$ - nonempty, closed, convex and bounded.
- $\mathbf{A} := [\mathbf{B}, \mathbf{C}]$.

The Fenchel dual problem

$$d^\star := \max_{\lambda \in \mathbb{R}^n} \left\{ d(\lambda) := -g^*_{\mathcal{U}}(-\mathbf{B}^T\lambda) - h^*_{\mathcal{V}}(-\mathbf{C}^T\lambda) + \langle \mathbf{b}, \lambda \rangle \right\}$$

- $g^*_{\mathcal{U}}$ and $h^*_{\mathcal{U}}$ are the Fenchel conjugate of $g_{\mathcal{U}} := g + \delta_{\mathcal{U}}$ and $h_{\mathcal{V}} := h + \delta_{\mathcal{V}}$, resp.

The dual function

$$d(\lambda) := \underbrace{\min_{\mathbf{u}\in\mathcal{U}} \left\{ g(\mathbf{u}) + \langle \mathbf{B}^T\lambda, \mathbf{u} \rangle \right\}}_{d^1(\lambda)} + \underbrace{\min_{\mathbf{v}\in\mathcal{V}} \left\{ h(\mathbf{v}) + \langle \mathbf{C}^T\lambda, \mathbf{v} \rangle \right\}}_{d^2(\lambda)} - \langle \mathbf{b}, \lambda \rangle.$$

[2] Q. Tran-Dinh and V. Cevher, *Splitting the Smoothed Primal-dual Gap: Optimal Alternating Direction Methods* Tech. Report, 2015, (http://arxiv.org/pdf/1507.03734.pdf) / (http://lions.epfl.ch/publications)

Standard ADMM as the dual Douglas-Rachford method

We can derive ADMM via the Douglas-Rachford splitting on the dual:

$$0 \in \mathbf{B}\partial g^*_{\mathcal{U}}(-\mathbf{B}^T\lambda) + \mathbf{C}\partial h^*{}_{\mathcal{V}}(-\mathbf{C}^T\lambda) + \boldsymbol{c},$$

which is the optimality condition of the **dual problem**.

Douglas-Rachford splitting method

$$\begin{cases} \mathbf{z}_g^k & := \mathrm{prox}_{\eta_k^{-1} g^*_{\mathcal{U}}(-\mathbf{B}^T\cdot)}(\lambda^k) \\ \mathbf{z}_h^k & := \mathrm{prox}_{\eta_k^{-1} h^*_{\mathcal{V}}(-\mathbf{C}^T\cdot)}(2\mathbf{z}_g^k - \lambda^k) \\ \lambda^{k+1} & := \lambda^k + (\mathbf{z}_g^k - \mathbf{z}_h^k). \end{cases}$$

Standard ADMM

$$\begin{cases} \mathbf{u}^{k+1} & := \arg\min_{\mathbf{u}\in\mathcal{U}} \left\{ g(\mathbf{u}) + \langle \lambda^k, \mathbf{B}\mathbf{u}\rangle + \frac{\eta_k}{2}\|\mathbf{B}\mathbf{u} + \mathbf{C}\mathbf{v}^k - \mathbf{b}\|^2 \right\} \\ \mathbf{v}^{k+1} & := \arg\min_{\mathbf{v}\in\mathcal{V}} \left\{ h(\mathbf{v}) + \langle \lambda^k, \mathbf{C}\mathbf{v}\rangle + \frac{\eta_k}{2}\|\mathbf{B}\mathbf{u}^{k+1} + \mathbf{C}\mathbf{v} - \mathbf{b}\|^2 \right\} \\ \lambda^{k+1} & := \lambda^k + \eta_k\left(\mathbf{B}\mathbf{u}^{k+1} + \mathbf{C}\mathbf{v}^{k+1} - \mathbf{b}\right). \end{cases}$$

Here, $\eta_k > 0$ is a given penalty parameter.

*Splitting the smoothed gap

Smoothing the gap

- The **dual components** d^1 and d^2 are nonsmooth. We smooth one, e.g., d^1, using:

$$d^1_\gamma(\lambda) := \min_{\mathbf{u}\in\mathcal{U}} \left\{ g(\mathbf{u}) + \frac{\gamma}{2}\|\mathbf{B}(\mathbf{u}-\mathbf{u}_c)\|^2 + \langle \lambda, \mathbf{B}\mathbf{u}\rangle \right\}$$

- Recall: We also approximate f by f_β as:

$$f_\beta(\mathbf{x}) := f(\mathbf{x}) + \frac{1}{2\beta}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2 \rightarrow f(\mathbf{x}) \ \text{ as } \mathbf{x} \text{ becomes feasible}$$

Three key properties of d^1_γ

- d^1_γ is concave and smooth.
- ∇d^1_γ is Lipschitz continuous with $L := \gamma^{-1}$.
- d^1_γ approximates d^1 as:

$$d^1_\gamma(\lambda) - \gamma D_{\mathcal{U}} \leq d^1(\lambda) \leq d^1_\gamma(\lambda),$$

where $D_{\mathcal{U}} := \max\left\{(1/2)\|\mathbf{B}(\mathbf{u}-\mathbf{u}_c)\|^2 : \mathbf{u}\in\mathcal{U}\right\}$.

*Our ADMM scheme: D-R on the smoothed gap

- Our new ADMM scheme consists of three steps:
ADMM step, acceleration step, and primal averaging.

Step 1: The main ADMM steps

$$\begin{cases} \hat{\mathbf{u}}^{k+1} & := \arg\min_{\mathbf{u}\in\mathcal{U}} \left\{ g_{\gamma_{k+1}}(\mathbf{u}) + \langle \hat{\lambda}^k, \mathbf{B}\mathbf{u}\rangle + \frac{\rho_k}{2}\|\mathbf{B}\mathbf{u} + \mathbf{C}\hat{\mathbf{v}}^k - \mathbf{b}\|^2 \right\} \\ \hat{\mathbf{v}}^{k+1} & := \arg\min_{\mathbf{v}\in\mathcal{V}} \left\{ h(\mathbf{v}) + \langle \hat{\lambda}^k, \mathbf{C}\mathbf{v}\rangle + \frac{\eta_k}{2}\|\mathbf{B}\hat{\mathbf{u}}^{k+1} + \mathbf{C}\mathbf{v} - \mathbf{b}\|^2 \right\} \\ \lambda^{k+1} & := \hat{\lambda}^k + \eta_k \left(\mathbf{B}\hat{\mathbf{u}}^{k+1} + \mathbf{C}\hat{\mathbf{v}}^{k+1} - \mathbf{b} \right). \end{cases}$$

where $g_\gamma(\cdot) := g(\cdot) + \frac{\gamma}{2}\|\mathbf{B}(\cdot - \mathbf{u}_c)\|^2$.

*The dual accelerated and primal averaging steps

- **Step 2:** [Dual acceleration] $\hat{\lambda}^k$ is computed as:

$$\hat{\lambda}^k := (1-\tau_k)\lambda_k + \frac{\tau_k}{\beta_k}(\mathbf{B}\mathbf{u}^k + \mathbf{C}\mathbf{v}^k - \mathbf{b}).$$

- **Step 3:** [Averaging] The primal iteration $\mathbf{x}^k := (\mathbf{u}^k, \mathbf{v}^k)$ is updated as:

$$\mathbf{u}^{k+1} := (1-\tau_k)\mathbf{u}^k + \tau_k\hat{\mathbf{u}}^{k+1} \quad \text{and} \quad \mathbf{v}^{k+1} := (1-\tau_k)\mathbf{v}^k + \tau_k\hat{\mathbf{v}}^{k+1}.$$

*How do we update parameters?

Duality gap and smoothed gap functions

- The duality gap: $G(\mathbf{w}) := f(\mathbf{x}) - d(\lambda)$, where $\mathbf{w} := (\mathbf{x}, \lambda)$.
- The smoothed gap: $\boxed{G_{\gamma\beta}(\mathbf{w}) := f_\beta(\mathbf{x}) - d_\gamma(\lambda)}$ with $d_\gamma := d^1_\gamma + d^2$.

Model-based gap reduction

The **gap reduction model** provides conditions to derive parameter update rules:

$$\boxed{G_{\gamma_{k+1}\beta_{k+1}}(\mathbf{w}^{k+1}) \leq (1-\tau_k)G_{\gamma_k\beta_k}(\mathbf{w}^k) + \tau_k(\eta_k + \rho_k)D_{\mathcal{X}}}$$

where $\gamma_{k+1} < \gamma_k$, $\beta_{k+1} < \beta_k$ and $D_{\mathcal{X}} := \max_{\mathbf{x}\in\mathcal{X}} \left\{(1/2)\|\mathbf{Bu} + \mathbf{Cv} - \mathbf{b}\|^2\right\}$.

Update rules

- **The smoothness parameters:** $\gamma_{k+1} := \frac{2\gamma_0}{k+3}$ and $\beta_k := \frac{9(k+3)}{\gamma_0(k+1)(k+7)}$.
- **The penalty parameters:** $\eta_k := \frac{\gamma_0}{k+3}$ and $\rho_k := \frac{3\gamma_0}{(k+3)(k+4)}$.
- **The step-size** $\tau_k := \frac{3}{k+4}$ $\Rightarrow$ $\mathcal{O}\left(\frac{1}{k}\right)$.

*Convergence guarantee & Other cases of interest

Convergence rate guarantee

- Rate on the primal objective residual and constraint feasibility:

$$f(\mathbf{x}^k) - f^\star \leq \frac{2\gamma_0 D_{\mathcal{U}}}{k+2} + \frac{3\gamma_0 D_{\mathcal{X}}}{2(k+3)}\left(1 + \frac{6}{k+2}\right) \Rightarrow \mathcal{O}\left(\frac{1}{k}\right)$$

$$\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\| \leq \frac{18 D_d^*}{\gamma_0(k+2)} + \frac{6}{k+2}\sqrt{D_{\mathcal{U}} + \frac{3(k+8)}{2(k+3)} D_{\mathcal{X}}} \Rightarrow \mathcal{O}\left(\frac{1}{k}\right)$$

where D_d^* is the diameter of the **dual solution set** $\Lambda^\star$.

- Lower bound: $-D_d^*\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\| \leq f(\mathbf{x}^k) - f^\star$.
- Rate on the dual objective residual:

$$d^\star - d(\lambda^k) \leq \frac{18(D_d^*)^2}{\gamma_0(k+2)} + \frac{6D_d^*}{k+2}\sqrt{D_{\mathcal{U}} + \frac{3(k+8)}{2(k+3)} D_{\mathcal{X}}} \Rightarrow \mathcal{O}\left(\frac{1}{k}\right).$$

Special cases: *cf.*, `http://lions.epfl.ch/publications`

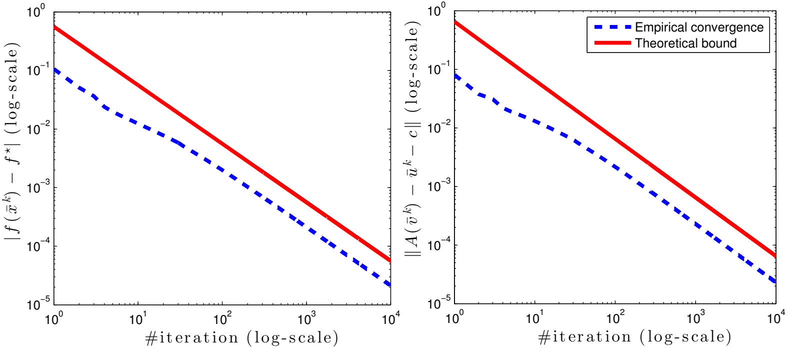
- Full-column rank or orthogonality of $\mathbf{A}$: Using smoothing term $(\gamma/2)\|\mathbf{u} - \mathbf{u}_c\|^2$.
- Strong convexity of g: We do not need to smooth d^1.
- Decomposability of g and $\mathcal{U}$: Using smoothing term

$$(\gamma/2)\sum_{i=1}^{s}\|\mathbf{B}_i(\mathbf{u}_i - \mathbf{u}_{c,i})\|^2.$$

*A comparison to the theoretical bounds

A stylized example: Square-root LASSO

$$f^{\star} := \min_{\mathbf{u}\in\mathcal{U},\mathbf{v}\in\mathcal{V}} \left\{ f(\mathbf{x}) := \|\mathbf{u}\|_2 + \kappa \|\mathbf{v}\|_1 : \mathbf{B}(\mathbf{v}) - \mathbf{u} = \boldsymbol{c} \right\}.$$



- See the preprint for more examples, enhancements, ...

References I

[1] Emmanuel J Candes, T. Strohmer, and V. Voroninski.
Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming.
IEEE Trans. Signal Processing, 60(5):2422–2432, 2012.

[2] David Gross, Yi-Kai Liu, Steven T. Flammia, Stephen R Becker, and Jens Eisert.
Quantum state tomography via compressed sensing.
Physical Review Letters, 105(15), 2010.

[3] Martin Jaggi.
Revisiting frank-wolfe: Projection-free sparse convex optimization.
In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, 2013.

[4] Yu. Nesterov.
Smooth minimization of non-smooth functions.
Math. Program., Ser. A, 103:127–152, 2005.

[5] Yu Nesterov.
Universal gradient methods for convex optimization problems.
Math. Program., 152:381–404, 2015.

References II

[6] Quoc Tran-Dinh and Volkan Cevher.
Constrained convex minimization via model-based excessive gap.
In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'14, 2014.

[7] Alp Yurtsever, Ya-Ping Hsieh, and Volkan Cevher.
Scalable convex methods for phase retrieval.
In *6th IEEE Intl. Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, 2015.

[8] Alp Yurtsever, Quoc Tran-Dinh, and Volkan Cevher.
A universal primal-dual convex optimization framework.
In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, 2015.