Fundamental trade-offs in primal-dual optimization

Volkan Cevher
volkan.cevher@epfl.ch

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

Simon Stevin Lecture

OPTEC–KU Leuven

[June 9, 2015]

Joint work with

Alp Yurtsever and Quoc Tran Dinh @ LIONS

lions@epfl

MARIE CURIE ACTIONS

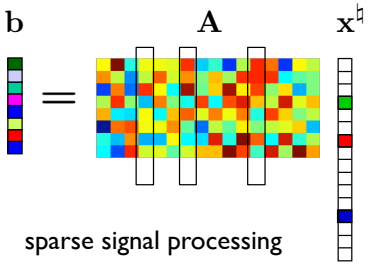
FNSNF
FONDS NATIONAL SUISSE
SCHWEIZERISCHER NATIONALFONDS
FONDO NAZIONALE SVIZZERO
SWISS NATIONAL SCIENCE FOUNDATION

erc

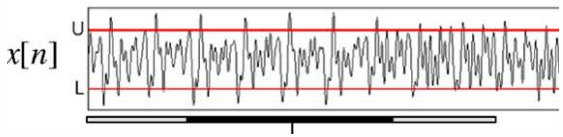
EPFL

Convex optimization in data sciences

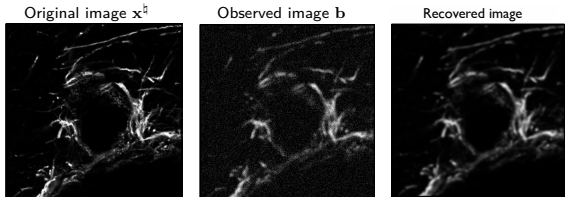
From SP and geophysics to medical and hyperspectral imaging



sparse signal processing



audio-signal processing



medical imaging

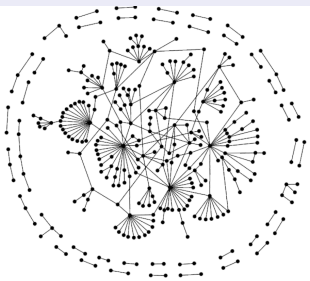
From machine learning and NLP to statistics and bioinformatics

NETFLIX

Recommendation systems

facebook

Topic models



Graphical model selection

From control and power systems to network science

Predictive control, system identification, controller design,...
Google page rank, social networks, transportation networks, power grids, utilities,...

Challenges for convex optimization

- High ambient dimension p and "big" data n
- Non-smooth objectives and constraint sets
- Increasingly elaborate observation models

Swiss army knife of convex formulations

A **primal problem** prototype

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

- 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 $\mathbf{x}^\star \in \mathcal{X}$

An example from the sparseland

$$\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ \|\mathbf{x}\|_1 : \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 \leq \kappa, \|\mathbf{x}\|_\infty \leq c \right\} \tag{SOCP}$$

Swiss army knife of convex formulations

A **primal problem** prototype

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

- 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 $\mathbf{x}^\star \in \mathcal{X}$

An example from the sparseland

$$\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ \|\mathbf{x}\|_1 : \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 \leq \kappa, \|\mathbf{x}\|_\infty \leq c \right\} \tag{SOCP}$$

Broader context for (1):

- Standard convex optimization formulations: *linear programming, convex quadratic programming, second order cone programming, semidefinite programming and geometric programming*.
- Reformulations of existing unconstrained problems via **convex splitting**: *composite convex minimization, consensus optimization, ...*

Swiss army knife of convex formulations

A **primal problem** prototype

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

- 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 $\mathbf{x}^\star \in \mathcal{X}$

A key advantage of the unified formulation (1): **Primal-dual methods**

- decentralized collection & storage of data
- distributed computation

Broader context for (1):

- Standard convex optimization formulations: *linear programming, convex quadratic programming, second order cone programming, semidefinite programming and geometric programming*.
- Reformulations of existing unconstrained problems via **convex splitting**: *composite convex minimization, consensus optimization, ...*

Primal-dual methods for (1): The zoo

- **Plenty . . .**

  cf., Lectures 7 and 8 @ `http://lions.epfl.ch/mathematics_of_data`

- Variants of the **Arrow-Hurwitz's method**:
  - Chambolle-Pock's algorithm [6], and its variants, e.g., He-Yuan's variant [20].
  - Primal-dual Hybrid Gradient (PDHG) method and its variants [15, 17].
  - Proximal-based decomposition (Chen-Teboulle's algorithm) [7].
- Splitting techniques from monotone inclusions:
  - Primal-dual splitting algorithms [3, 5, 8, 33, 9, 10].
  - Three-operator splitting [11].
- Dual splitting techniques:
  - Alternating minimization algorithms (AMA) [16, 33].
  - Alternating direction methods of multipliers (ADMM) [13, 21].
  - Accelerated variants of AMA and ADMM [10, 18].
  - Preconditioned ADMM, Linearized ADMM and inexact Uzawa algorithms [6, 29].
- **Second-order decomposition methods:**
  - Dual (quasi) Newton methods [34].
  - Smoothing decomposition methods via barriers functions [24, 32, 36].

Performance of optimization algorithms

Exact vs. approximate solutions

- Computing an **exact solution** $\mathbf{x}^\star$ to (1) is **impracticable**
- Algorithms seek $\mathbf{x}^\star_\epsilon$ that approximates $\mathbf{x}^\star$ up to ϵ in some sense

A performance metric: Time-to-reach ϵ

`time-to-reach` ϵ `= number of iterations to reach` ϵ $\times$ `per iteration time`

Performance of optimization algorithms

Exact vs. approximate solutions

- Computing an **exact solution** $\mathbf{x}^\star$ to (1) is **impracticable**
- Algorithms seek $\mathbf{x}^\star_\epsilon$ that approximates $\mathbf{x}^\star$ up to ϵ in some sense

A performance metric: Time-to-reach ϵ

`time-to-reach` ϵ `= number of iterations to reach` ϵ $\times$ `per iteration time`

A key issue:

The notion of ϵ-accuracy is elusive in constrained optimization!

Numerical ϵ-accuracy

- **Unconstrained case:** All iterates are feasible (no advantage from infeasibility)!

$$f(\mathbf{x}_\epsilon^\star) - f^\star \leq \epsilon$$

$$\boxed{f^\star = \min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x})}$$

Numerical ϵ-accuracy

- **Unconstrained case:** All iterates are feasible (no advantage from infeasibility)!

$$f(\mathbf{x}_\epsilon^\star) - f^\star \leq \epsilon$$

- **Constrained case:** We need to also measure the infeasibility of the iterates!

$$f^\star - f(\mathbf{x}_\epsilon^\star) \leq \epsilon \quad !!!$$

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

Numerical ϵ-accuracy

- **Unconstrained case:** All iterates are feasible (no advantage from infeasibility)!

$$f(\mathbf{x}_\epsilon^\star) - f^\star \leq \epsilon$$

- **Constrained case:** We need to also measure the infeasibility of the iterates!

$$f^\star - f(\mathbf{x}_\epsilon^\star) \leq \epsilon \quad !!!$$

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

Our definition of ϵ-accurate solutions [30]

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

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

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

Numerical ϵ-accuracy

- **Unconstrained case:** All iterates are feasible (no advantage from infeasibility)!

$$f(\mathbf{x}_\epsilon^\star) - f^\star \leq \epsilon$$

- **Constrained case:** We need to also measure the infeasibility of the iterates!

$$f^\star - f(\mathbf{x}_\epsilon^\star) \leq \epsilon \quad !!!$$

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

Our definition of ϵ-accurate solutions [30]

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

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

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

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

Performance of optimization algorithms

A performance metric: Time-to-reach ϵ

`time-to-reach` ϵ `= number of iterations to reach` ϵ $\times$ `per iteration time`

Finding the fastest algorithm within the zoo is tricky!

- heuristics & tuning parameters
- non-optimal rates & strict assumptions
- lack of precise characterizations

Performance of optimization algorithms

A performance metric: Time-to-reach ϵ

time-to-reach ϵ = number of iterations to reach ϵ $\times$ per iteration time

Finding the fastest algorithm within the zoo is tricky!

- heuristics & tuning parameters
- non-optimal rates & strict assumptions
- lack of precise characterizations

This talk:

tuning-free and optimal algorithms with precise characterizations and key trade-offs!

Outline

The proximal way

The sharp way

Conclusions

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 $\Rightarrow$ Find $\lambda^\star$ and then solve for $\mathbf{x}^\star=\mathbf{x}^*(\lambda^\star)$

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 $\Rightarrow$ Find $\lambda^\star$ and then solve for $\mathbf{x}^\star=\mathbf{x}^*(\lambda^\star)$

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 $\lambda^\star_{\bar{\epsilon}}\rightarrow\mathbf{x}^\star_\epsilon$ (i.e., $\bar{\epsilon}(\epsilon)$), where ϵ is for the original constrained problem (1)

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 $\Rightarrow$ Find $\lambda^\star$ and then solve for $\mathbf{x}^\star=\mathbf{x}^*(\lambda^\star)$

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 $\lambda^\star_{\bar{\epsilon}}\to\mathbf{x}^\star_\epsilon$ (i.e., $\bar{\epsilon}(\epsilon)$), where ϵ is for the original constrained problem (1)

Efficiency considerations for the dual problem

Subgradient method

1. Choose $\lambda^0 \in \mathbb{R}^n$.
2. For $k = 0, 1, \cdots$, perform:
$$\lambda^{k+1} = \lambda^k + \alpha_k \mathbf{v}^k,$$
where $\mathbf{v}^k \in \partial d(\lambda^k)$ and α_k is the step-size.

Subgradient method for the dual

Assume that the following conditions

1. $\|\mathbf{v}\|_2 \leq G$ for all $\mathbf{v} \in \partial d(\lambda)$, $\lambda \in \mathbb{R}^n$.
2. $\|\lambda^0 - \lambda^\star\|_2 \leq R$

Let the step-size be chosen as $\alpha_k = \frac{R}{G\sqrt{k}}$.
Then, the subgradient method satisfies

$$\min_{0 \leq i \leq k} d^\star - d(\lambda^i) \leq \frac{RG}{\sqrt{k}}$$

Efficiency considerations for the dual problem

Subgradient method

1. Choose $\lambda^0 \in \mathbb{R}^n$.
2. For $k = 0, 1, \cdots$, perform:
$$\lambda^{k+1} = \lambda^k + \alpha_k \mathbf{v}^k,$$
where $\mathbf{v}^k \in \partial d(\lambda^k)$ and α_k is the step-size.

Subgradient method for the dual

Assume that the following conditions

1. $\|\mathbf{v}\|_2 \leq G$ for all $\mathbf{v} \in \partial d(\lambda)$, $\lambda \in \mathbb{R}^n$.
2. $\|\lambda^0 - \lambda^\star\|_2 \leq R$

Let the step-size be chosen as $\alpha_k = \frac{R}{G\sqrt{k}}$.
Then, the subgradient method satisfies

$$\min_{0 \leq i \leq k} d^\star - d(\lambda^i) \leq \frac{RG}{\sqrt{k}} \leq \bar{\epsilon}$$

SGM: $\mathcal{O}\left(\frac{1}{\bar{\epsilon}^2}\right) \times$ subgradient calculation

Efficiency considerations for the dual problem

Gradient method

1. Choose $\lambda^0 \in \mathbb{R}^n$.
2. For $k = 0, 1, \cdots$, perform:
$$\lambda^{k+1} = \lambda^k + \frac{1}{L}\nabla d(\lambda^k),$$

where L is the Lipschitz constant.

Subgradient method for the dual

Assume that the following conditions

1. $\|\mathbf{v}\|_2 \leq G$ for all $\mathbf{v} \in \partial d(\lambda)$, $\lambda \in \mathbb{R}^n$.
2. $\|\lambda^0 - \lambda^\star\|_2 \leq R$

Let the step-size be chosen as $\alpha_k = \frac{R}{G\sqrt{k}}$.
Then, the subgradient method satisfies

$$\min_{0\leq i\leq k} d^\star - d(\lambda^i) \leq \frac{RG}{\sqrt{k}} \leq \bar{\epsilon}$$

SGM: $\mathcal{O}\left(\frac{1}{\bar{\epsilon}^2}\right)$ $\times$ subgradient calculation

GM: $\mathcal{O}\left(\frac{1}{\bar{\epsilon}}\right)$ $\times$ gradient calculation

Impact of smoothness

(Lipschitz gradient) $d(\lambda)$ has Lipschitz continuous gradient iff

$$\|\nabla d(\lambda) - \nabla d(\eta)\|_2 \leq L\|\lambda - \eta\|_2$$

for all $\lambda, \eta \in \text{dom}(d)$ and we indicate this structure as $d(\lambda) \in \mathcal{F}_L$.

For all $d(\lambda) \in \mathcal{F}_L$, the gradient method with step-size $1/L$ obeys

$$d^\star - d(\lambda^k) \leq \frac{2LR^2}{k+4} \leq \bar{\epsilon}.$$

Efficiency considerations for the dual problem

Gradient method

1. Choose $\lambda^0 \in \mathbb{R}^n$.
2. For $k = 0, 1, \cdots$, perform:
$$\lambda^{k+1} = \lambda^k + \frac{1}{L}\nabla d(\lambda^k),$$

where L is the Lipschitz constant.

Subgradient method for the dual

Assume that the following conditions

1. $\|\mathbf{v}\|_2 \leq G$ for all $\mathbf{v} \in \partial d(\lambda)$, $\lambda \in \mathbb{R}^n$.
2. $\|\lambda^0 - \lambda^\star\|_2 \leq R$

Let the step-size be chosen as $\alpha_k = \frac{R}{G\sqrt{k}}$.
Then, the subgradient method satisfies

$$\min_{0 \leq i \leq k} d^\star - d(\lambda^i) \leq \frac{RG}{\sqrt{k}} \leq \bar{\epsilon}$$

SGM: $\mathcal{O}\left(\frac{1}{\bar{\epsilon}^2}\right)$ $\times$ subgradient calculation

GM: $\mathcal{O}\left(\frac{1}{\bar{\epsilon}}\right)$ $\times$ gradient calculation

Impact of smoothness

(Lipschitz gradient) $d(\lambda)$ has Lipschitz continuous gradient iff

$$\|\nabla d(\lambda) - \nabla d(\eta)\|_2 \leq L\|\lambda - \eta\|_2$$

for all $\lambda, \eta \in \mathrm{dom}(d)$ and we indicate this structure as $d(\lambda) \in \mathcal{F}_L$.

For all $d(\lambda) \in \mathcal{F}_L$, the gradient method with step-size $1/L$ obeys

$$d^\star - d(\lambda^k) \leq \frac{2LR^2}{k+4} \leq \bar{\epsilon}.$$

This is NOT the best we can do.

There exists a complexity lower-bound

$$d^\star - d(\lambda^k) \geq \frac{3LR^2}{32(k+1)^2}, \forall d(\lambda) \in \mathcal{F}_L,$$

for any iterative method based only on function and gradient evaluations.

Efficiency considerations for the dual problem

Accelerated gradient method

1. Choose $\mathbf{u}^0 = \lambda^0 \in \mathbb{R}^n$.
2. For $k = 0, 1, \cdots$, perform:

$$\lambda^k = \mathbf{u}^k + \frac{1}{L}\nabla d(\mathbf{u}^k),$$
$$\mathbf{u}^{k+1} = \lambda^k + \rho_k(\lambda^k - \lambda^{k-1}),$$

where L is the Lipschitz constant, and ρ_k is a momentum parameter.

Subgradient method for the dual

Assume that the following conditions

1. $\|\mathbf{v}\|_2 \leq G$ for all $\mathbf{v} \in \partial d(\lambda)$, $\lambda \in \mathbb{R}^n$.
2. $\|\lambda^0 - \lambda^\star\|_2 \leq R$

Let the step-size be chosen as $\alpha_k = \frac{R}{G\sqrt{k}}$.
Then, the subgradient method satisfies

$$\min_{0\leq i\leq k} d^\star - d(\lambda^i) \leq \frac{RG}{\sqrt{k}} \leq \bar{\epsilon}$$

SGM: $\mathcal{O}\left(\frac{1}{\bar{\epsilon}^2}\right)$ $\times$ subgradient calculation

GM: $\mathcal{O}\left(\frac{1}{\bar{\epsilon}}\right)$ $\times$ gradient calculation

AGM: $\mathcal{O}\left(\frac{1}{\sqrt{\bar{\epsilon}}}\right)$ $\times$ gradient calculation

Impact of smoothness

(Lipschitz gradient) $d(\lambda)$ has Lipschitz continuous gradient iff

$$\|\nabla d(\lambda) - \nabla d(\eta)\|_2 \leq L\|\lambda - \eta\|_2$$

for all $\lambda, \eta \in \text{dom}(d)$ and we indicate this structure as $d(\lambda) \in \mathcal{F}_L$.

For all $d(\lambda) \in \mathcal{F}_L$, the accelerated gradient method with momentum $\rho_k = \frac{k+1}{k+3}$ obeys

$$d^\star - d(\lambda^k) \leq \frac{2LR^2}{(k+2)^2} \leq \bar{\epsilon}$$

This is NEARLY the best we can do.

There exists a complexity lower-bound

$$d^\star - d(\lambda^k) \geq \frac{3LR^2}{32(k+1)^2}, \forall d(\lambda) \in \mathcal{F}_L,$$

for any iterative method based only on function and gradient evaluations.

Number of iterations: From $\mathcal{O}\left(\frac{1}{\bar{\epsilon}^2}\right)$ to $\mathcal{O}\left(\frac{1}{\bar{\epsilon}}\right)$

When can the dual function have Lipschitz gradient?

When $f(\mathbf{x})$ is γ-strongly convex, the dual function $d(\lambda)$ is $\frac{\|\mathbf{A}\|^2}{\gamma}$-Lipschitz gradient.

(Strong convexity) $f(\mathbf{x})$ is γ-strongly convex iff $f(\mathbf{x}) - \frac{\gamma}{2}\|\mathbf{x}\|_2^2$ is convex.

$$d(\lambda) = \min_{\mathbf{x}:\mathbf{x}\in\mathcal{X}} \underbrace{f(\mathbf{x}) - \frac{\gamma}{2}\|\mathbf{x}\|_2^2}_{\substack{\text{convex \&} \\ \text{possibly nonsmooth}}} + \langle \lambda, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle + \underbrace{\frac{\gamma}{2}\|\mathbf{x}\|_2^2}_{\text{leads to } d\in\mathcal{F}_L}$$

AGM automatically obtains $d^\star - d(\mathbf{x}^k) \leq \bar{\epsilon}$ with $k = \mathcal{O}\left(\frac{1}{\sqrt{\bar{\epsilon}}}\right)$

Number of iterations: From $\mathcal{O}\left(\frac{1}{\bar{\epsilon}^2}\right)$ to $\mathcal{O}\left(\frac{1}{\bar{\epsilon}}\right)$

When can the dual function have Lipschitz gradient?

When $f(\mathbf{x})$ is γ-strongly convex, the dual function $d(\lambda)$ is $\frac{\|\mathbf{A}\|^2}{\gamma}$-Lipschitz gradient.

(Strong convexity) $f(\mathbf{x})$ is γ-strongly convex iff $f(\mathbf{x}) - \frac{\gamma}{2}\|\mathbf{x}\|_2^2$ is convex.

$$d(\lambda) = \min_{\mathbf{x}:\mathbf{x}\in\mathcal{X}} \underbrace{f(\mathbf{x}) - \frac{\gamma}{2}\|\mathbf{x}\|_2^2}_{\text{convex \& possibly nonsmooth}} + \langle \lambda, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle + \underbrace{\frac{\gamma}{2}\|\mathbf{x}\|_2^2}_{\text{leads to } d\in\mathcal{F}_L}$$

Our idea: Apply Nesterov's smoothing [28] to the dual

$$\boxed{d_\gamma(\lambda) = \min_{\mathbf{x}:\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) + \langle \lambda, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle + \frac{\gamma}{2}\|\mathbf{x}\|_2^2}$$

1. $\nabla d_\gamma(\lambda) = \mathbf{A}\mathbf{x}_\gamma^*(\lambda) - \mathbf{b}$
2. $d_\gamma(\lambda) - \gamma\mathcal{D}_\mathcal{X} \leq d(\lambda) \leq d_\gamma(\lambda)$, where $\mathcal{D}_\mathcal{X} = \max_{\mathbf{x}\in\mathcal{X}} \frac{1}{2}\|\mathbf{x}\|_2^2$.
3. λ^k of AGM on $d_\gamma(\lambda)$ has $d^\star - d(\lambda^k) \leq \gamma\mathcal{D}_\mathcal{X} + d_\gamma^\star - d_\gamma(\lambda^k) \leq \gamma\mathcal{D}_\mathcal{X} + \frac{2\|\mathbf{A}\|^2 R^2}{\gamma(k+2)^2}$.
4. We minimize the upperbound wrt γ and obtain $d^\star - d(\lambda^k) \leq \bar{\epsilon}$ with $k = \mathcal{O}\left(\frac{1}{\bar{\epsilon}}\right)$.

Per-iteration time: The key role of the prox-operator

Smoothed dual: $d_\gamma(\lambda) = \min_{\mathbf{x}:\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) + \langle \lambda, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle + \frac{\gamma}{2}\|\mathbf{x}\|_2^2$

$$\mathbf{x}^*(\lambda) := \mathrm{prox}^{\mathcal{X}}_{f/\gamma}\left(-\frac{1}{\gamma}\mathbf{A}^T\lambda\right)$$

Per-iteration time: The key role of the prox-operator

Smoothed dual: $d_\gamma(\lambda) = \min_{\mathbf{x}:\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) + \langle \lambda, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle + \frac{\gamma}{2}\|\mathbf{x}\|_2^2$

$$\mathbf{x}^*(\lambda) := \operatorname{prox}^{\mathcal{X}}_{f/\gamma}\left(-\frac{1}{\gamma}\mathbf{A}^T\lambda\right)$$

Definition (Prox-operator)

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

Key properties:

- single valued & non-expansive.
- distributes when the primal problem has decomposable structure:

$$f(\mathbf{x}) := \sum_{i=1}^{m} f_i(\mathbf{x}_i), \quad \text{and} \quad \mathcal{X} := \mathcal{X}_1 \times \cdots \times \mathcal{X}_m.$$

  where $m \geq 1$ is the number of components.
- often efficient & has closed form expression. For instance, if $f(\mathbf{z}) = \|\mathbf{z}\|_1$, then the prox-operator performs coordinate-wise soft-thresholding by 1.

Going from the dual $\bar{\epsilon}$ to the primal ϵ–I

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 $\lambda^\star_{\bar{\epsilon}} \to \mathbf{x}^\star_\epsilon$ (i.e., $\bar{\epsilon}(\epsilon)$), where ϵ is for the original constrained problem (1)

Going from the dual $\bar{\epsilon}$ to the primal ϵ–I

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 $\lambda^\star_{\bar{\epsilon}} \to \mathbf{x}^\star_\epsilon$ (i.e., $\bar{\epsilon}(\epsilon)$), where ϵ is for the original constrained problem (1)

Measuring progress via the gap function

We can define a **gap function** to measure our progress for $\mathbf{z} := (\mathbf{x}, \lambda) \in \mathcal{X} \times \mathbb{R}^n$

$$G(\mathbf{z}) = \underbrace{\max_{\hat{\lambda} \in \mathbb{R}^n} f(\mathbf{x}) + \langle \hat{\lambda}, \mathbf{A}\mathbf{x} - \mathbf{b} \rangle}_{= f(\mathbf{x}) \text{ if } \mathbf{A}\mathbf{x} = \mathbf{b}, \infty \text{ o/w}} - \underbrace{\min_{\hat{\mathbf{x}} \in \mathcal{X}} f(\hat{\mathbf{x}}) + \langle \lambda, \mathbf{A}\hat{\mathbf{x}} - \mathbf{b} \rangle}_{= d(\lambda)} \geq 0$$

- $G(\mathbf{z}^\star) = 0$ iff $\mathbf{z}^\star := (\mathbf{x}^\star, \lambda^\star)$ is a primal-dual solution of (1).
- Primal accuracy ϵ and the dual accuracy $\bar{\epsilon}$ can be related via the gap function.

Going from the dual $\bar{\epsilon}$ to the primal ϵ–II

A smoothed gap function measuring the primal-dual gap

We define a smoothed version of the gap function

$$G_{\gamma\beta}(\mathbf{z}) = \underbrace{\max_{\hat{\lambda}\in\mathbb{R}^n} f(\mathbf{x}) + \langle \hat{\lambda}, \mathbf{A}\mathbf{x} - \mathbf{b}\rangle - \frac{\beta}{2}\|\hat{\lambda}\|_2^2}_{f_\beta(\mathbf{x}) = f(\mathbf{x}) + \frac{1}{2\beta}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2} - \underbrace{\min_{\hat{\mathbf{x}}\in\mathcal{X}} f(\hat{\mathbf{x}}) + \langle \lambda, \mathbf{A}\hat{\mathbf{x}} - \mathbf{b}\rangle + \frac{\gamma}{2}\|\hat{\mathbf{x}}\|_2^2}_{d_\gamma(\lambda)}$$

Going from the dual $\bar{\epsilon}$ to the primal ϵ–II

A smoothed gap function measuring the primal-dual gap

We define a smoothed version of the gap function

$$G_{\gamma\beta}(\mathbf{z}) = \underbrace{\max_{\hat{\lambda}\in\mathbb{R}^n} f(\mathbf{x}) + \langle \hat{\lambda}, \mathbf{A}\mathbf{x}-\mathbf{b}\rangle - \frac{\beta}{2}\|\hat{\lambda}\|_2^2}_{f_\beta(\mathbf{x})=f(\mathbf{x})+\frac{1}{2\beta}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2} - \underbrace{\min_{\hat{\mathbf{x}}\in\mathcal{X}} f(\hat{\mathbf{x}}) + \langle \lambda, \mathbf{A}\hat{\mathbf{x}}-\mathbf{b}\rangle + \frac{\gamma}{2}\|\hat{\mathbf{x}}\|_2^2}_{d_\gamma(\lambda)}$$

Our new technique: Model-based gap reduction MGR (cf., [30])

Let $G_k(\cdot) := G_{\gamma_k\beta_k}(\cdot)$. We generate a **sequence** $\{\mathbf{z}^k, \gamma_k, \beta_k\}_{k\geq 0}$ such that

$$\boxed{G_{k+1}(\mathbf{z}^{k+1}) \leq (1-\tau_k)\, G_k(\mathbf{z}^k) + \psi_k} \qquad \text{(MGR)}$$

for $\psi_k \to 0$, rate $\tau_k \in (0,1)$ $(\sum_k \tau_k = \infty)$, $\gamma_k\beta_{k+1} < \gamma_k\beta_k$ so that $G_{\gamma_k\beta_k}(\cdot) \to G(\cdot)$.

- **Consequence**: $\boxed{G(\mathbf{z}^k) \to 0^+ \quad \Rightarrow \quad \mathbf{z}^k \to \mathbf{z}^\star = (\mathbf{x}^\star, \lambda^\star)}$ (primal-dual solution).

MGR ties $\bar{\epsilon}$ to ϵ via $f_\beta(\mathbf{x})$

An instance of our primal-dual scheme

The standard scheme ([26])

The accelerated scheme for maximing $d_\gamma \in \mathcal{F}_L^{1,1}$ consists of three main steps:

$$\begin{cases} \hat{\lambda}^k & := (1-\tau_k)\lambda^k + \tau_k \tilde{\lambda}_k \\ \lambda^{k+1} & := \hat{\lambda}^k + \frac{1}{L_{d_\gamma}} \nabla d_\gamma(\hat{\lambda}^k) \\ \tilde{\lambda}_{k+1} & := \lambda_k^* - \frac{1}{\tau_k}(\hat{\lambda}^k - \lambda^{k+1}). \end{cases} \tag{2}$$

Here, L_{d_γ} is the Lipschitz constant of ∇d_γ and $\tau_k \in (0,1)$ is a given momentum term.

Our primal-dual scheme (`http://lions.epfl.ch/decopt`)

Our approach is fundamentally the same as the accelerated gradient method:

$$\begin{cases} \hat{\lambda}^k & := (1-\tau_k)\lambda^k + \tau_k \tilde{\lambda}^k \\ \lambda^{k+1} & := \hat{\lambda}^k + \frac{\gamma_{k+1}}{\|\mathbf{A}\|^2}(\mathbf{A}\mathbf{x}^*_{\gamma_{k+1}}(\hat{\lambda}^k) - \mathbf{b}) \\ \mathbf{x}^{k+1} & := (1-\tau_k)\mathbf{x}^k + \tau_k \mathbf{x}^*_{\gamma_{k+1}}(\hat{\lambda}^k) \\ \tilde{\lambda}^{k+1} & := \frac{1}{\beta_{k+1}}(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}). \end{cases} \tag{3}$$

Both smoothing parameters γ and β are updated at each iteration.

Going from the dual $\bar{\epsilon}$ to the primal ϵ–III

An uncertainty relation via MGR ([31, 30])

The product of the primal and dual convergence rates is lowerbounded by MGR:

$$\boxed{\gamma_k \beta_k \geq \frac{\tau_k^2}{1-\tau_k^2} \|\mathbf{A}\|^2}$$

Note that $\tau_k^2 = \Omega\left(\frac{1}{k^2}\right)$ due to Nesterov's lowerbound.

- The rate of γ_k controls the primal residual: $|f(\mathbf{x}^k) - f^\star| \leq \mathcal{O}(\gamma_k)$
- The rate of β_k controls the feasibility: $\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|_2 \leq \mathcal{O}(\beta_k + \tau_k) = \mathcal{O}(\beta_k)$
- They cannot be simultaneously $\mathcal{O}\left(\frac{1}{k^2}\right)$!

Convergence guarantee

Theorem [31, 30]

1. When f is non-smooth, the best we can do is $\gamma_k = \mathcal{O}\left(\frac{1}{k}\right)$ and $\beta_k = \mathcal{O}\left(\frac{1}{k}\right)$:

$$\begin{cases} -D_{\Lambda^\star}\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\| \leq & f(\mathbf{x}^k) - f^\star \leq \frac{C_p D_{\mathcal{X}}}{k+1}, \\ & \|\mathbf{A}\mathbf{x}^k - \mathbf{b}\| \leq \frac{C_d(D_{\Lambda^\star} + \sqrt{D_{\mathcal{X}}})}{k+1}, \end{cases}$$

   where C_p and C_d are two given positive constants depending on different schemes.

2. When f is strongly convex with $\mu > 0$, we can take $\gamma_k = \mu$ and $\beta_k = \mathcal{O}\left(\frac{1}{k^2}\right)$:

$$\begin{cases} -D_{\Lambda^\star}\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\| \leq & f(\mathbf{x}^k) - f^\star \leq 0 \\ & \|\mathbf{A}\mathbf{x}^k - \mathbf{b}\| \leq \frac{4\|\mathbf{A}\|^2}{(k+2)^2\mu} D_{\Lambda^\star} \\ & \|\mathbf{x}^k - \mathbf{x}^\star\| \leq \frac{4\|\mathbf{A}\|}{(k+2)\mu} D_{\Lambda^\star} \end{cases}$$

   where $D_{\Lambda^\star} := \min\{\|\lambda^\star\| \ : \ \lambda^\star \in \Lambda^\star\}$ the **norm** of the min-norm dual solution.

Example: An application of the convergence guarantees

Problem (Consensus optimization)

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}) \right\}$$

Constrained reformulation via a product space trick with $\mathbf{z} := [\mathbf{x}_1, \ldots, \mathbf{x}_n]$*:*

$$F^\star := \min_{\mathbf{z}:=[\mathbf{x}_1,\ldots,\mathbf{x}_n]\in\mathbb{R}^{np}} \left\{ F(\mathbf{z}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}_i) : \mathbf{x}_i - \mathbf{x}_j = 0, (i,j) \in E \right\}$$

for some user-defined graph $\mathcal{G} = (V, E)$ *with vertices* V *and edges* E*.*

Example: An application of the convergence guarantees

Problem (Consensus optimization)

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}) \right\}$$

Constrained reformulation via a product space trick with $\mathbf{z} := [\mathbf{x}_1, \ldots, \mathbf{x}_n]$*:*

$$F^\star := \min_{\mathbf{z}:=[\mathbf{x}_1,\ldots,\mathbf{x}_n]\in\mathbb{R}^{np}} \left\{ F(\mathbf{z}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}_i) : \mathbf{x}_i - \mathbf{x}_j = 0, (i,j) \in E \right\}$$

for some user-defined graph $\mathcal{G} = (V, E)$ *with vertices* V *and edges* E*.*

Interpretation of the convergence guarantees

By using our algorithm in a decentralized but synchronized fashion, we obtain

$$|F(\mathbf{z}^k) - f^\star| \leq \mathcal{O}(1/k) \quad \text{and} \quad \sum_{(i,j)\in E} \|\mathbf{x}_i^k - \mathbf{x}_j^k\|^2 \leq \mathcal{O}(1/k^2), \;\; i = 1, \ldots, n-1.$$

If f_i's are strongly convex, these rates further improve.

Number of iterations: From $\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$ to $\mathcal{O}\left(\frac{1}{\sqrt{\epsilon}}\right)$

The augmented Lagrangian (AL) smoothing

$$d_\gamma(\lambda) := \min_{\mathbf{x}\in\mathcal{X}} \left\{ f(\mathbf{x}) + \langle \lambda, \mathbf{A}\mathbf{x} - \mathbf{b} \rangle + \frac{\gamma}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 \right\}$$

1. d_γ now has Lipschitz continuous gradient with $L_{d_\gamma} = \gamma^{-1}$.
2. $\nabla d_\gamma(\lambda) = \mathbf{A}\mathbf{x}^*_\gamma(\lambda) - \mathbf{b}$.
3. $\mathbf{x}^*_\gamma(\lambda)$ can be computed approximately by first-order methods.

Augmented Lagrangian idea: The trade-offs

An uncertainty relation via MGR

The product of the primal and dual convergence rates is lowerbounded by MGR:

$$\gamma\beta_{k+1} \geq \tau_k^2.$$

Here, we update β_k as $\beta_{k+1} = (1-\tau_k)\beta_k$. Then $\beta_k = \Omega(\tau_k^2)$.
Note that $\tau_k^2 = \Omega\left(\frac{1}{k^2}\right)$ due to Nesterov's lowerbound.

- The rate of β_k controls the primal residual: $|f(\mathbf{x}^k) - f^\star| \leq \mathcal{O}(\beta_k)$
- The rate of β_k controls the feasibility: $\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|_2 \leq \mathcal{O}(\beta_k)$
- They can be simultaneously $\mathcal{O}\left(\frac{1}{k^2}\right)$!

No free lunch: Large γ increases the difficulty of per-iteration time!

Augmented Lagrangian idea: The trade-offs

Theorem (convergence guarantee) [31, 30]

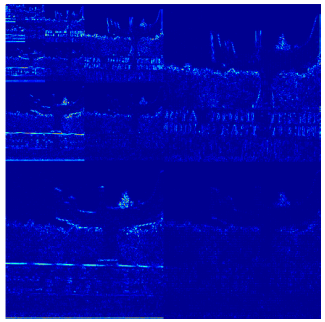
The sequence $\{\mathbf{z}^k\}$ generated by our accelerated scheme satisfies:

$$-\tfrac{\gamma}{2}\|\mathbf{A}\mathbf{x}^k-\mathbf{b}\|^2-\|\mathbf{A}\mathbf{x}^k-\mathbf{b}\|D_{\Lambda^\star}\leq \quad f(\mathbf{x}^k)-f^\star \quad \leq 0$$
$$\|\mathbf{A}\mathbf{x}^k-\mathbf{b}\| \quad \leq \tfrac{8D_{\Lambda^\star}}{\gamma(k+1)^2}.$$

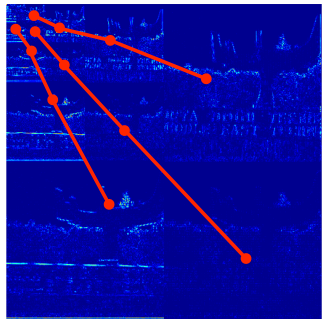
The worst-case iteration complexity: $\mathcal{O}\left(\sqrt{\frac{D_{\Lambda^\star}}{\gamma\epsilon}}\right)$.

- We can increase γ to obtain faster convergence
- However, it becomes more difficult to compute $\mathbf{x}^*_\gamma(\hat{\lambda}^k)$!
- Warm starts help but we need to solve subproblems with increasing accuracy!

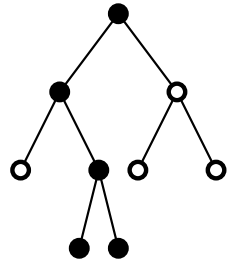
Tree sparsity [23, 12, 2, 37]



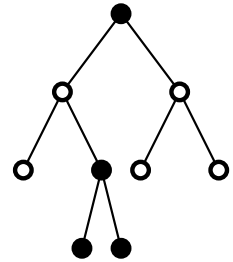
Wavelet coefficients



Wavelet tree

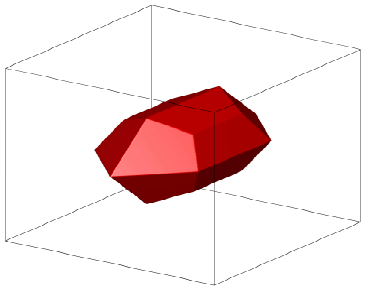


Valid selection of nodes

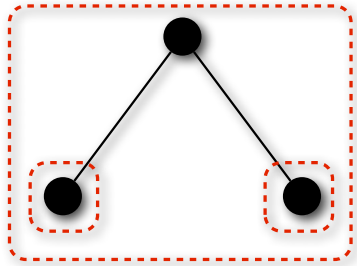


Invalid selection of nodes

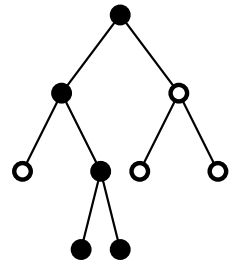
Tree sparsity [23, 12, 2, 37]



$f(\mathbf{x})$-ball



$\mathfrak{G} = \{\{1,2,3\},\{2\},\{3\}\}$



valid selection of nodes

Structure: *We seek the sparsest signal with a rooted connected subtree support.*

Compressive sensing formulation (TU-relax [14])

$$\begin{array}{ll} \min\limits_{\mathbf{x}\in\mathbb{R}^p} & f(\mathbf{x}) := \sum\nolimits_{\mathcal{G}_i\in\mathfrak{G}} \|\mathbf{x}_{\mathcal{G}_i}\|_\infty \\ \text{s.t.} & \mathbf{A}\mathbf{x} = \mathbf{b}. \end{array} \tag{4}$$

This problem possesses two key structures: decomposability and tractable proximity.

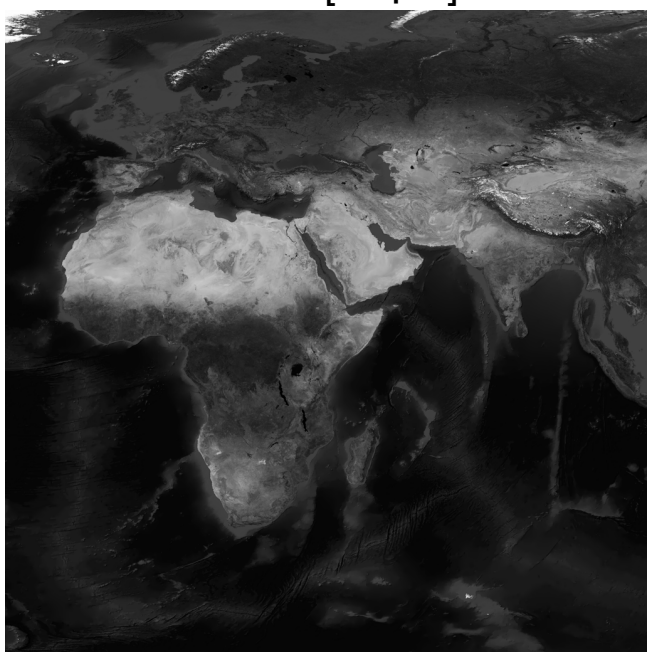
When $g = p$ and $\mathcal{G}_i = \{i\}$, (4) reduces to the well-known basis pursuit (BP):

$$\min_{\mathbf{x}\in\mathbb{R}^p} \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \mathbf{A}\mathbf{x} = \mathbf{b}. \tag{5}$$

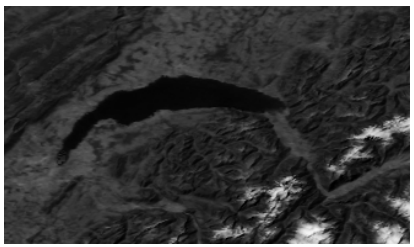
Tree sparsity example: 1:100-compressive sensing

$$(n, p) = (10^7, 10^9)$$

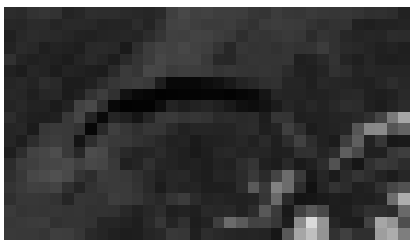
World [1Gpix]



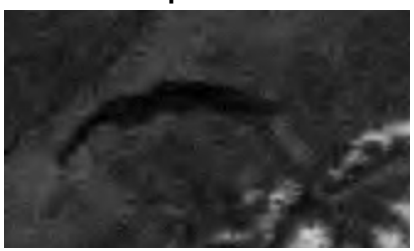
Lac Léman



World [10Mpix]

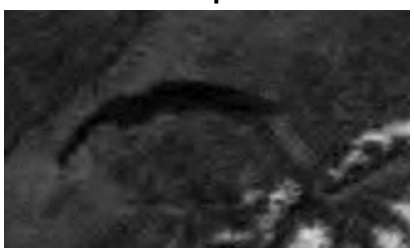


sparse



PNSR $= 31.83$db

tree-sparse



PNSR $= 32.48$db

Sampling: Breaking the coherence barrier [1]

Recovery: Augmented Lagrangian method [31]

Iterations: 113
PD gap: 1e−8
Applications of $(\mathbf{A}, \mathbf{A}^T)$: $(684, 570)$

Outline

The proximal way

The sharp way

Conclusions

Revisiting the prox-operator

Prox-operator helps us process nonsmooth terms "efficiently"

$$\mathrm{prox}_f^{\mathcal{X}}(\mathbf{x}) := \operatorname*{argmin}_{\mathbf{z}\in\mathcal{X}} \{f(\mathbf{z}) + (1/2)\|\mathbf{z}-\mathbf{x}\|^2\}.$$

Key properties:

- single valued & non-expansive.
- distributes when the problem has decomposable structure:

$$f(\mathbf{x}) := \sum_{i=1}^{m} f_i(\mathbf{x}_i), \quad \text{and} \quad \mathcal{X} := \mathcal{X}_1 \times \cdots \times \mathcal{X}_m.$$

  where $m \geq 1$ is the number of components.
- often efficient & has closed form expression. For instance, if $f(\mathbf{z}) = \|\mathbf{z}\|_1$ and $\mathcal{X} = \mathbb{R}^p$, then the prox-operator performs coordinate-wise soft-thresholding by 1.

Revisiting the prox-operator

Prox-operator helps us process nonsmooth terms "efficiently"

$$\text{prox}_f^{\mathcal{X}}(\mathbf{x}) := \underset{\mathbf{z}\in\mathcal{X}}{\text{argmin}}\{f(\mathbf{z}) + (1/2)\|\mathbf{z}-\mathbf{x}\|^2\}.$$

Key properties:

- single valued & non-expansive.
- distributes when the problem has decomposable structure:

$$f(\mathbf{x}) := \sum_{i=1}^{m} f_i(\mathbf{x}_i), \quad \text{and} \quad \mathcal{X} := \mathcal{X}_1 \times \cdots \times \mathcal{X}_m.$$

  where $m \geq 1$ is the number of components.
- often efficient & has closed form expression. For instance, if $f(\mathbf{z}) = \|\mathbf{z}\|_1$ and $\mathcal{X} = \mathbb{R}^p$, then the prox-operator performs coordinate-wise soft-thresholding by 1.

Not all nonsmooth functions are proximal-friendly!

If $g(\mathbf{z}) = \|\mathbf{z}\|_\star$ (i.e., the nuclear norm of $\mathbf{z}$) then

- No decomposition!
- the prox-operator <> full singular value decomposition!

Revisiting the prox-operator

Prox-operator helps us process nonsmooth terms "efficiently"

$$\text{prox}_f^{\mathcal{X}}(\mathbf{x}) := \underset{\mathbf{z}\in\mathcal{X}}{\text{argmin}}\{f(\mathbf{z}) + (1/2)\|\mathbf{z}-\mathbf{x}\|^2\}.$$

Key properties:

- single valued & non-expansive.
- distributes when the problem has decomposable structure:

$$f(\mathbf{x}) := \sum_{i=1}^{m} f_i(\mathbf{x}_i), \quad \text{and} \quad \mathcal{X} := \mathcal{X}_1 \times \cdots \times \mathcal{X}_m.$$

  where $m \geq 1$ is the number of components.
- often efficient & has closed form expression. For instance, if $f(\mathbf{z}) = \|\mathbf{z}\|_1$ and $\mathcal{X} = \mathbb{R}^p$, then the prox-operator performs coordinate-wise soft-thresholding by 1.

Not all nonsmooth functions are proximal-friendly!

If $g(\mathbf{z}) = \|\mathbf{z}\|_\star$ (i.e., the nuclear norm of $\mathbf{z}$) then

- No decomposition!
- the prox-operator <> full singular value decomposition!

Can we avoid the prox-operator for something cheaper?

Frank-Wolfe's method: Earliest example

Problem setting

$$\boxed{f^\star := \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})} \tag{6}$$

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: Earliest example

Problem setting

$$\boxed{f^\star := \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})} \tag{6}$$

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 [22] for a review)

Conditional gradient method (CGA)

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

$$\boxed{f^\star := \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})} \tag{6}$$

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 [22] for a review)

Conditional gradient method (CGA)

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 $\mathcal{X} := \{\mathbf{x} \in \mathbb{R}^{n\times p} : \|\mathbf{x}\|_\star \leq 1\}$, $(*)$ corresponds to rank-1 updates!

Towards Fenchel-type operators

Generalized sharp operators [35]

We define the (generalized) sharp operator of a convex function g over $\mathcal{X}$ as follows

$$\boxed{[\mathbf{x}]^{\sharp}_{\mathcal{X},g} := \operatorname*{argmax}_{\mathbf{z}\in\mathcal{X}} \{\langle \mathbf{x}, \mathbf{z}\rangle - g(\mathbf{z})\}\,.}$$

Important special cases:

1. If $g = 0$, then we obtain the so-called linear minimization oracle.
2. If $\mathcal{X} = \mathrm{dom}(g)$, then $[\mathbf{x}]^{\sharp}_{g} = \nabla g^*(\mathbf{x})$, where g^* is the Fenchel conjugate of g.

Towards Fenchel-type operators

Generalized sharp operators [35]

We define the (generalized) sharp operator of a convex function g over $\mathcal{X}$ as follows

$$\boxed{[\mathbf{x}]^\sharp_{\mathcal{X},g} := \operatorname*{argmax}_{\mathbf{z}\in\mathcal{X}} \{\langle \mathbf{x}, \mathbf{z}\rangle - g(\mathbf{z})\}\,.}$$

Important special cases:

1. If $g = 0$, then we obtain the so-called linear minimization oracle.
2. If $\mathcal{X} = \operatorname{dom}(g)$, then $[\mathbf{x}]^\sharp_g = \nabla g^*(\mathbf{x})$, where g^* is the Fenchel conjugate of g.

Example (Nuclear norm)

Consider $g(\mathbf{x}) := \frac{1}{2}\|\mathbf{x}\|^2_\star$ and $\mathcal{X} := \{\mathbf{x} \in \mathbb{R}^{n\times p} : \|\mathbf{x}\|_\star \leq 1\}$. Let $\mathbf{u}$ and $\mathbf{v}$ be the left and right principal singular vectors of $\mathbf{x}$ respectively. Then,

$$\mathbf{u}\mathbf{v}^T \in [\mathbf{x}]^\sharp_{\mathcal{X}} := [\mathbf{x}]^\sharp_{\mathcal{X},0}\,, \qquad \|\mathbf{x}\|\mathbf{u}\mathbf{v}^T \in [\mathbf{x}]^\sharp_g := [\mathbf{x}]^\sharp_{\mathbb{R}^{n\times p},g}$$

where $\|\cdot\|$ is the spectral norm. **The computations are essentially the same.**

Revisiting Frank-Wolfe's method

Problem setting

$$\boxed{f^\star := \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})} \tag{7}$$

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 [22] for a review)

Conditional gradient method (CGA)

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

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

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

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

Generalized conditional gradient method replaces the indicator function $\delta_{\mathcal{X}}$ with g:

$$\hat{\mathbf{x}}^k := \arg\min_{\mathbf{x}} \{g(\mathbf{x}) + \nabla f(\mathbf{x}^k)^T \mathbf{x}\} = [\nabla f(\mathbf{x}^k)]_g^\sharp.$$

Revisiting Frank-Wolfe's method

Problem setting

$$\boxed{f^\star := \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})} \tag{7}$$

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 $\mathbf{Ax} - \mathbf{b} \in \mathcal{K}$ and nonsmooth $f(\mathbf{x})$ with the sharp-operator

Frank-Wolfe's method (see [22] for a review)

Conditional gradient method (CGA)

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

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

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

Generalized conditional gradient method replaces the indicator function $\delta_{\mathcal{X}}$ with g:

$$\hat{\mathbf{x}}^k := \arg\min_{\mathbf{x}} \{g(\mathbf{x}) + \nabla f(\mathbf{x}^k)^T \mathbf{x}\} = [\nabla f(\mathbf{x}^k)]_g^\sharp.$$

Brief detour: Exploring the smoothness in depth

Definition (Hölder continuous gradients [25])

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

A convex function g has Hölder continuous subgradients of degree ν if there are nonnegative real constants $\nu \in [0,1]$ and M_ν that satisfy:

$$\|\nabla g(\mathbf{x}) - \nabla g(\mathbf{y})\|_* \leq M_\nu \|\mathbf{x}-\mathbf{y}\|^\nu$$

where $\nabla g(\lambda)$ is a (sub)gradient of g.

Brief detour: Exploring the smoothness in depth

Definition (Hölder continuous gradients [25])

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

A convex function g has Hölder continuous subgradients of degree ν if there are nonnegative real constants $\nu \in [0,1]$ and M_ν that satisfy:

$$\|\nabla g(\mathbf{x}) - \nabla g(\mathbf{y})\|_* \leq M_\nu \|\mathbf{x}-\mathbf{y}\|^\nu$$

where $\nabla g(\lambda)$ is a (sub)gradient of g.

Highlights

1. $\nu = 1$ is the Lipschitz continuous gradients case where $L = M_\nu$.
2. $\nu = 0$ is the bounded gradient assumption (recall the subgradient method).
3. Iteration lowerbound for the Hölder class: $\mathcal{O}\left(\left(\frac{M_\nu \|\mathbf{x}^0 - \mathbf{x}^\star\|^{1+\nu}}{\epsilon} \right)^{\frac{2}{1+3\nu}} \right)$.
4. The condition also ensures the following basic surrogate for any $\mathbf{x}, \mathbf{y} \in \mathcal{X}$:

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

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:

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

where we use 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 methods [27]

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

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

- 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 [27] 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:

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

where we use 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 methods [27]

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

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

- 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 [27].

[1] PGM in [27] uses the Bregman / prox setup.

Universal primal-dual gradient methods

Our strategy: Hölder smoothness in the dual

Instead of smoothing, we assume that d is Hölder continuous for some $\nu \in [0, 1]$:

$$M_\nu(d) := \sup_{\lambda \neq \eta} \frac{\|\nabla d(\lambda) - \nabla d(\eta)\|_*}{\|\lambda - \eta\|^\nu}, \qquad M_d^* := \inf_{0 \leq \nu \leq 1} M_\nu(d) < +\infty.$$

In the sequel:

- Solve the dual problem by a new FISTA algorithm [4]
- Develop new primal strategies to approximate $\mathbf{x}^\star$
- Bound the total number of linesearch steps

Universal primal-dual gradient methods

Our strategy: Hölder smoothness in the dual

Instead of smoothing, we assume that d is Hölder continuous for some $\nu \in [0, 1]$:

$$M_\nu(d) := \sup_{\lambda \neq \eta} \frac{\|\nabla d(\lambda) - \nabla d(\eta)\|_*}{\|\lambda - \eta\|^\nu}, \qquad M_d^* := \inf_{0 \leq \nu \leq 1} M_\nu(d) < +\infty.$$

In the sequel:

- Solve the dual problem by a new FISTA algorithm [4]
- Develop new primal strategies to approximate $\mathbf{x}^\star$
- Bound the total number of linesearch steps

Is this assumption reasonable?

- If $\mathcal{X}$ is bounded and d is subdifferentiable, then ∇d is also bounded.
- If f is uniformly convex with convexity parameter $\mu_f > 0$ and degree $q \geq 2$, i.e.,

$$\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \mu_f \|\mathbf{x} - \mathbf{y}\|^q$$

for any $\mathbf{x}, \mathbf{y} \in \mathcal{X}$, then ∇d satisfies Hölder condition with $\nu = \frac{1}{q-1}$ and $M_\nu = \left(\mu_f^{-1} \|\mathbf{A}\|^2\right)^{\frac{1}{q-1}}$.

Our universal primal-dual gradient methods: The dual steps

Dual steps: Characterized by dual gradient updates [35]

$$\mathbf{x}^*(\lambda^k) := \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) + \langle \mathbf{A}^T\lambda^k, \mathbf{x}\rangle \right\} \equiv \left[-\mathbf{A}^T\lambda^k \right]_f^\sharp$$

- **(UniPDGrad)** requires 2 linesearch steps on the average with ϵ:

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

- **(AccUniPDGrad)** requires 1 linesearch step on the average with ϵ/t_k:

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

The primal steps

Primal steps: Characterized by weighted averaging [35]

$$\boxed{\mathbf{x}^*(\lambda^k) := \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) + \langle \mathbf{A}^T\lambda^k, \mathbf{x}\rangle \right\} \equiv \left[-\mathbf{A}^T\lambda^k\right]_f^\sharp}$$

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

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

Number of iterations

The worst-case iteration complexity [35]

To achieve $\bar{\mathbf{x}}^k$ such that $|f(\bar{\mathbf{x}}^k) - f^\star| \leq \epsilon$ and $\|\mathbf{A}\bar{\mathbf{x}}^k - \mathbf{b}\| \leq \epsilon$ is:

$$\begin{cases} \text{For } \textbf{(UniPDGrad):} & \mathcal{O}\left(D_{\Lambda^\star}^2 \inf_{0\leq\nu\leq 1} \left(\frac{M_\nu}{\epsilon}\right)^{\frac{2}{1+\nu}} \right), \quad \text{(optimal for } \nu = 0\text{)}. \\ \text{For } \textbf{(AccUniPDGrad):} & \mathcal{O}\left(D_{\Lambda^\star}^{\frac{2+5\nu}{1+3\nu}} \inf_{0\leq\nu\leq 1} \left(\frac{M_\nu}{\epsilon}\right)^{\frac{2}{1+3\nu}} \right), \quad \text{(optimal for } \nu = 1\text{)}. \end{cases}$$

Number of iterations

The worst-case iteration complexity [35]

To achieve $\bar{\mathbf{x}}^k$ such that $|f(\bar{\mathbf{x}}^k) - f^\star| \leq \epsilon$ and $\|\mathbf{A}\bar{\mathbf{x}}^k - \mathbf{b}\| \leq \epsilon$ is:

$$\begin{cases} \text{For } \textbf{(UniPDGrad)}: & \mathcal{O}\left(D_{\Lambda^\star}^2 \inf_{0\leq\nu\leq 1} \left(\frac{M_\nu}{\epsilon}\right)^{\frac{2}{1+\nu}} \right), \quad \text{(optimal for } \nu = 0\text{)}. \\ \text{For } \textbf{(AccUniPDGrad)}: & \mathcal{O}\left(D_{\Lambda^\star}^{\frac{2+5\nu}{1+3\nu}} \inf_{0\leq\nu\leq 1} \left(\frac{M_\nu}{\epsilon}\right)^{\frac{2}{1+3\nu}} \right), \quad \text{(optimal for } \nu = 1\text{)}. \end{cases}$$

The sharp-operator vs. the prox-operator

No free lunch. When $d(\lambda) = \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) + \langle \mathbf{A}^T\lambda, \mathbf{x}\rangle \right\}$ is fully nonsmooth,

- **(UniPDGrad)** with the sharp-operator obtains $\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$
- Our PD decomposition methods with the prox operator obtain $\mathcal{O}\left(\frac{1}{\epsilon}\right)$

Overall efficiency depends on the per-iteration time (i.e., complexity of prox vs. sharp).

Scalability 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 $\mathcal{A}$ [19], we can deduce the state from $\mathbf{b} = \mathcal{A}(\mathbf{X}) \in \mathcal{C}^n$ based on the following convex optimization formulation:

$$\varphi^\star := \min_{\mathbf{X} \in \mathcal{S}_+^p} \left\{ \frac{1}{2} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|_2^2 : \mathrm{tr}(\mathbf{X}) = 1 \right\}. \tag{8}$$

The recovery is also robust to noise.

Scalability 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 $\mathcal{A}$ [19], we can deduce the state from $\mathbf{b} = \mathcal{A}(\mathbf{X}) \in \mathcal{C}^n$ based on the following convex optimization formulation:

$$\varphi^\star := \min_{\mathbf{X} \in \mathcal{S}_+^p} \left\{ \frac{1}{2} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|_2^2 : \mathrm{tr}(\mathbf{X}) = 1 \right\}. \tag{8}$$

The recovery is also robust to noise.

Perfect scalability test: tuning free constraint + Lipschitz continuous gradient

Setup [35]

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 $\epsilon = 2 \cdot 10^{-4}$.

Scalability example: Quantum tomography with Pauli operators - II

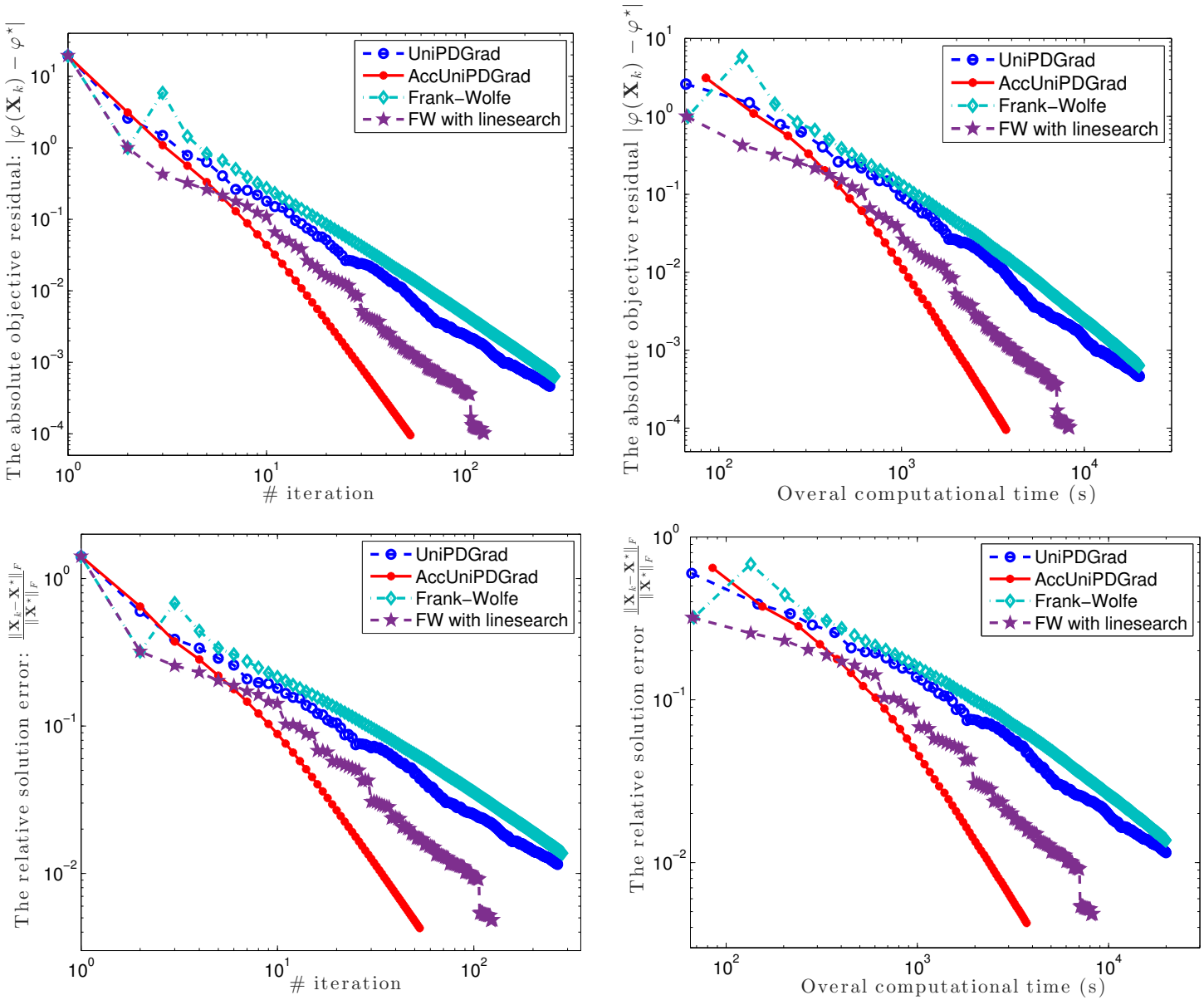


Figure: The performance of (Acc)UniProx and Frank-Wolfe algorithms for (13).

Flexibility example: Matrix completion with MovieLens dataset - I

Problem formulation

Let $\Omega \subseteq \{1, \cdots, p\} \times \{1, \cdots, q\}$ be a subset of indexes and $\mathbf{M}_\Omega = (\mathbf{M}_{ij})_{(i,j)\in\Omega}$ be the observed entries of a missed matrix $\mathbf{M}$. $\mathcal{P}_\Omega$ is the projection on the subset Ω.

$$f^\star := \min_{\mathbf{X}\in\mathbb{R}^{p\times q}} \left\{ \frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{X}) - \mathbf{M}_\Omega\|_2^2 : \|\mathbf{X}\|_\star \leq \kappa \right\} \tag{9}$$

We can also solve another robust version against outliers:

$$f^\star := \min_{\mathbf{X}\in\mathbb{R}^{p\times q}} \left\{ \frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{X}) - \mathbf{M}_\Omega\|_1^2 : \|\mathbf{X}\|_\star \leq \kappa \right\}. \tag{10}$$

Note that Frank-Wolfe cannot solve (10).

Flexibility example: Matrix completion with MovieLens dataset - II

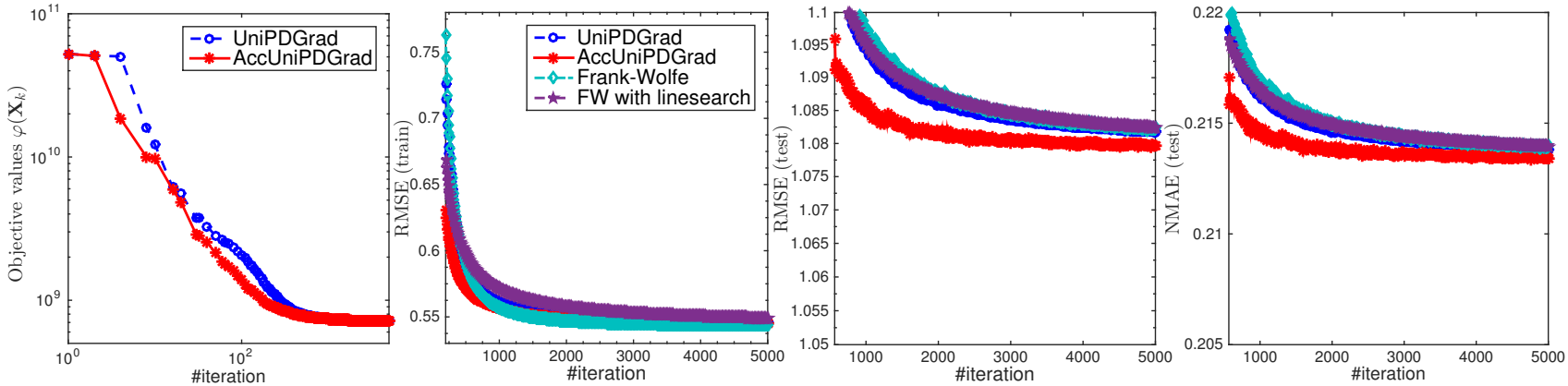


Figure: The performance of UniProx and AccUniProx algorithms for (9) and (10).

Setup [35]

- MovieLens $100k$ dataset: $100'000$ ratings from 943 users on 1682 movies
- Input parameters $\lambda^0 = \mathbf{0}^n$, $\epsilon = 2 \cdot 10^{-2}$ and $\kappa = 9975/2$.

The general constraint case

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

the universal dual accelerated gradient method:

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

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:

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

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

Flexibility example II: Matrix completion with MovieLens dataset

Problem formulation

Let $\Omega \subseteq \{1, \cdots, p\} \times \{1, \cdots, q\}$ be a subset of indexes and $\mathbf{M}_\Omega = (\mathbf{M}_{ij})_{(i,j)\in\Omega}$ be the observed entries of a missed matrix $\mathbf{M}$. $\mathcal{P}_\Omega$ is the projection on the subset Ω.

$$f^\star := \min_{\mathbf{X}\in\mathbb{R}^{p\times q}} \left\{ \frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{X}) - \mathbf{M}_\Omega\|_2^2 : \|\mathbf{X}\|_\star \leq \kappa \right\} \tag{11}$$

We can also solve another robust version against outliers:

$$f^\star := \min_{\mathbf{X}\in\mathbb{R}^{p\times q}} \left\{ \frac{1}{2}\|\mathcal{P}_\Omega(\mathbf{X}) - \mathbf{M}_\Omega\|_1^2 : \|\mathbf{X}\|_\star \leq \kappa \right\}. \tag{12}$$

Note that Frank-Wolfe cannot solve (12).

Problem formulation

Following formulation may be easier to tune with an expected perturbation level τ:

$$f^\star := \min_{\mathbf{X}\in\mathbb{R}^{p\times q}} \left\{ \frac{1}{2}\|\mathbf{X}\|_\star^2 : \|\mathcal{P}_\Omega(\mathbf{X}) - \mathbf{M}_\Omega\|_1 \leq \tau \right\}. \tag{13}$$

Note that Frank-Wolfe cannot solve (13)

Flexibility example II: Matrix completion with MovieLens dataset

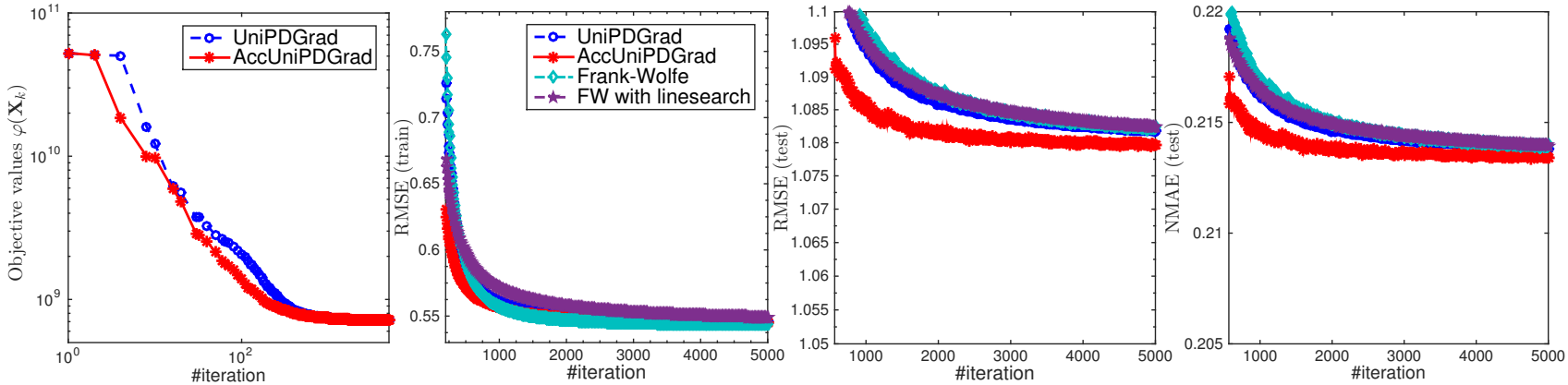


Figure: The performance of UniProx and AccUniProx algorithms for (11) and (12).

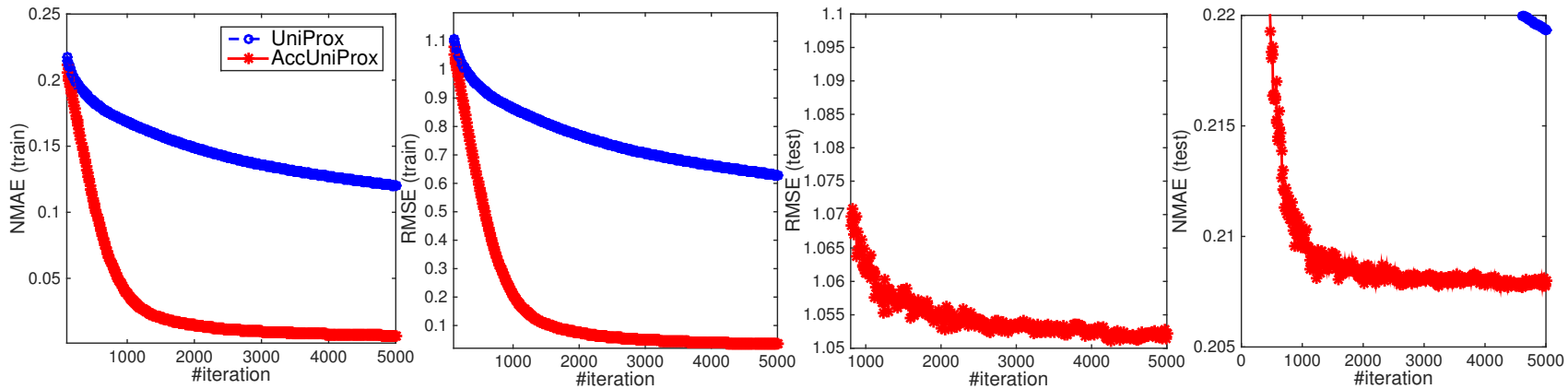


Figure: The performance of UniProx and AccUniProx algorithms for (13).

Setup [35]

- $\kappa = 9975/2.$
- $\tau = 4\times$ NMAE $\times$# of test samples

Outline

The proximal way

The sharp way

Conclusions

Conclusions

- Our contributions for the prototype primal problem (1):
  - new proximal primal dual algorithms via the model-based gap reduction technique
  - new universal primal-dual algorithms via sharp operators
  - guarantees on primal optimality and feasibility of the iterates for all
- Uncertainty relations for the proximal way
  - primal optimality of the algorithmic iterates competes with their feasibility
  - clear benefits of the augmented Lagrangian
- The prox-operator vs. the sharp-operator
  - tradeoffs abound (also along the Augmented Lagrangian + ADMM / AMA)
  - time-winner is unclear due to modern data processing systems
  - warm-starts / hybrid strategies ?

References I

[1] Ben Adcock, Anders C. Hansen, Clarice Poon, and Bogdan Roman.
Breaking the coherence barrier: A new theory for compressed sensing.
http://arxiv.org/abs/1302.0561, Feb. 2013.

[2] R.G. Baraniuk, V. Cevher, M.F. Duarte, and C. Hegde.
Model-based compressive sensing.
Information Theory, IEEE Transactions on, 56(4):1982–2001, 2010.

[3] H.H. Bauschke and P. Combettes.
Convex analysis and monotone operators theory in Hilbert spaces.
Springer-Verlag, 2011.

[4] A. Beck and M. Teboulle.
A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems.
SIAM J. Imaging Sciences, 2(1):183–202, 2009.

[5] Luis M Briceno-Arias and Patrick L Combettes.
A monotone+ skew splitting model for composite monotone inclusions in duality.
SIAM Journal on Optimization, 21(4):1230–1250, 2011.

[6] A. Chambolle and T. Pock.
A first-order primal-dual algorithm for convex problems with applications to imaging.
Journal of Mathematical Imaging and Vision, 40(1):120–145, 2011.

References II

[7] G. Chen and M. Teboulle.
A proximal-based decomposition method for convex minimization problems.
Math. Program., 64:81–101, 1994.

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

[9] D. Davis.
Convergence rate analysis of the forward-Douglas-Rachford splitting scheme.
UCLA CAM report 14-73, 2014.

[10] D. Davis and W. Yin.
Faster convergence rates of relaxed Peaceman-Rachford and ADMM under regularity assumptions.
UCLA CAM report 14-58, 2014.

[11] D. Davis and W. Yin.
A three-operator splitting scheme and its optimization applications.
Tech. Report., 2015.

References III

[12] Marco F. Duarte, Dharmpal Davenport, Mark A. adn Takhar, Jason N. Laska, Ting Sun, Kevin F. Kelly, and Richard G. Baraniuk.
Single-pixel imaging via compressive sampling.
IEEE Sig. Process. Mag., 25(2):83–91, March 2008.

[13] J. Eckstein and D. Bertsekas.
On the Douglas - Rachford splitting method and the proximal point algorithm for maximal monotone operators.
Math. Program., 55:293–318, 1992.

[14] Marwa El Halabi and Volkan Cevher.
A totally unimodular view of structured sparsity.
In *18th Int. Conf. Artificial Intelligence and Statistics*, 2015.

[15] J. E. Esser.
Primal-dual algorithm for convex models and applications to image restoration, registration and nonlocal inpainting.
Phd. thesis, University of California, Los Angeles, Los Angeles, USA, 2010.

[16] D. Gabay and B. Mercier.
A dual algorithm for the solution of nonlinear variational problems via finite element approximation.
Computers & Mathematics with Applications, 2(1):17 – 40, 1976.

References IV

[17] T. Goldstein, E. Esser, and R. Baraniuk.
Adaptive Primal-Dual Hybrid Gradient Methods for Saddle Point Problems.
Tech. Report., http://arxiv.org/pdf/1305.0546v1.pdf:1–26, 2013.

[18] T. Goldstein, B. ODonoghue, and S. Setzer.
Fast Alternating Direction Optimization Methods.
SIAM J. Imaging Sci., 7(3):1588–1623, 2012.

[19] D. Gross, Y.-K. Liu, S. T. Flammia, S. Becker, and J. Eisert.
Quantum state tomography via compressed sensing.
Physical Review Letters, 105(15):150401(0–4), 2010.

[20] B. He and X. Yuan.
Convergence analysis of primal-dual algorithms for saddle-point problem: from contraction perspective.
SIAM J. Imaging Sciences, 5:119–149, 2012.

[21] B.S. He and X.M. Yuan.
On the $O(1/n)$ convergence rate of the Douglas-Rachford alternating direction method.
SIAM J. Numer. Anal., 50:700–709, 2012.

References V

[22] M. Jaggi.
Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization.
JMLR W&CP, 28(1):427–435, 2013.

[23] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach.
Proximal methods for hierarchical sparse coding.
J. Mach. Learn. Res., 12:2297–2334, 2011.

[24] I. Necoara and J.A.K. Suykens.
Interior-point lagrangian decomposition method for separable convex optimization.
J. Optim. Theory and Appl., 143(3):567–588, 2009.

[25] A. S. Nemirovsky and D. B. Yudin.
Problem complexity and method efficiency in optimization.
John Wiley & Sons, Chichester, 1983.

[26] Y. Nesterov.
A method for unconstrained convex minimization problem with the rate of convergence $o(1/k^2)$.
Doklady AN SSSR, 269(translated as Soviet Math. Dokl.):543–547, 1983.

[27] Y. Nesterov.
Universal gradient methods for convex optimization problems.
Math. Program., xx:1–24, 2014.

References VI

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

[29] Y. Ouyang, Y. Chen, G. LanG. Lan., and E. JR. Pasiliao.
An accelerated linearized alternating direction method of multiplier.
Tech, 2014.

[30] Q. Tran-Dinh and V. Cevher.
Constrained convex minimization via model-based excessive gap.
In *Proc. the Neural Information Processing Systems Foundation conference (NIPS2014)*, pages 1–9, Montreal, Canada, December 2014.

[31] Q. Tran-Dinh and V. Cevher.
A primal-dual algorithmic framework for constrained convex minimization.
Tech. Report., LIONS, pages 1–54, 2014.

[32] Q. Tran-Dinh, I. Necoara, C. Savorgnan, and M. Diehl.
An Inexact Perturbed Path-Following Method for Lagrangian Decomposition in Large-Scale Separable Convex Optimization.
SIAM J. Optim., 23(1):95–125, 2013.

References VII

[33] P. Tseng.
Applications of splitting algorithm to decomposition in convex programming and variational inequalities.
SIAM J. Control Optim., 29:119–138, 1991.

[34] E. Wei, A. Ozdaglar, and A.Jadbabaie.
A Distributed Newton Method for Network Utility Maximization.
http://web.mit.edu/asuman/www/publications.htm, 2011.

[35] A. Yurtsever, Q. Tran-Dinh, and V. Cevher.
Universal primal-dual proximal-gradient methods.
Tech. Report. (LIONS, EPFL), Available at:
http://arxiv.org/pdf/1502.03123.pdf., 2015.

[36] G. Zhao.
A Lagrangian dual method with self-concordant barriers for multistage stochastic convex programming.
Math. Progam., 102:1–24, 2005.

[37] Peng Zhao, Guilherme Rocha, and Bin Yu.
Grouped and hierarchical model selection through composite absolute penalties.
Department of Statistics, UC Berkeley, Tech. Rep, 703, 2006.