Mathematics of Data: From Theory to Computation

Prof. Volkan Cevher
volkan.cevher@epfl.ch

Lecture 12: Primal-dual optimization II

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

EE-556 (Fall 2020)

License Information for Mathematics of Data Slides

Outline

- This class:
  1. Algorithms for solving min-max optimization
- Next class
  1. Additional scalable optimization methods for constrained minimization

A roadmap to algorithms for convex-concave minimax optimization

Recall: A restricted minimax formulation

Let us consider

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y}), \tag{1}$$

where $\Phi(\mathbf{x},\mathbf{y})$ is convex in $\mathbf{x}$ and concave in $\mathbf{y}$.

○ In the sequel, we consider the following cases

1. $\mathcal{X} \subset \mathbb{R}^p$ and $\mathcal{Y} \subset \mathbb{R}^n$; and $\Phi(\mathbf{x},\mathbf{y})$ is smooth, or bilinear, or strongly convex/strongly concave
   - Algorithms: Proximal-Point [18], Extra-gradient [8, 13, 12], OGDA [13, 12]
2. $\mathcal{X} \subset \mathbb{R}^p$ and $\mathcal{Y} \subset \mathbb{R}^n$ with tractable "mirror maps"; and $\Phi(\mathbf{x},\mathbf{y})$ is smooth and continuously differentiable
   - Algorithm: Mirror-Prox [14]
3. $\mathcal{X} = \mathbb{R}^p$ and $\mathcal{Y} = \mathbb{R}^n$; and $\Phi(\mathbf{x},\mathbf{y}) = h(\mathbf{x}) + f(\mathbf{x}) + \langle \mathbf{A}\mathbf{x},\mathbf{y}\rangle - g^*(\mathbf{y})$
   - Algorithms: Chambolle-Pock [4], Condat-Vu [5, 21], PD3O [23]

Smooth unconstrained minimax optimization

Details of the restricted minimax formulation

$$\min_{\mathbf{x}\in\mathbb{R}^d}\max_{\mathbf{y}\in\mathbb{R}^n}\Phi(\mathbf{x},\mathbf{y}).$$

We assume that

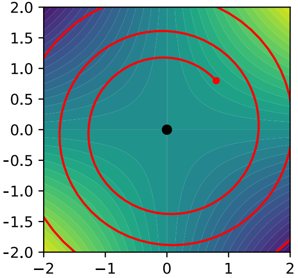
- $\Phi(\cdot,\mathbf{y})$ is convex for all $\mathbf{y}\in\mathbb{R}^n$,
- $\Phi(\mathbf{x},\cdot)$ is concave for all $\mathbf{x}\in\mathbb{R}^d$,
- $\Phi(\mathbf{x},\mathbf{y})$ is continuously differentiable in $\mathbf{x}$ and $\mathbf{y}$,
- Φ is smooth in the following sense.

$$\|\mathbf{V}(\mathbf{z_1})-\mathbf{V}(\mathbf{z_2})\| := \left\| \begin{bmatrix} \nabla_{\mathbf{x}}\Phi(\mathbf{x}_1,\mathbf{y}_1) \\ -\nabla_{\mathbf{y}}\Phi(\mathbf{x}_1,\mathbf{y}_1) \end{bmatrix} - \begin{bmatrix} \nabla_{\mathbf{x}}\Phi(\mathbf{x}_2,\mathbf{y}_2) \\ -\nabla_{\mathbf{y}}\Phi(\mathbf{x}_2,\mathbf{y}_2) \end{bmatrix} \right\| \le L \left\| \begin{bmatrix} \mathbf{x}_1-\mathbf{x}_2 \\ \mathbf{y}_1-\mathbf{y}_2 \end{bmatrix} \right\|, \text{where} \quad \mathbf{z}=\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix} \quad (2)$$

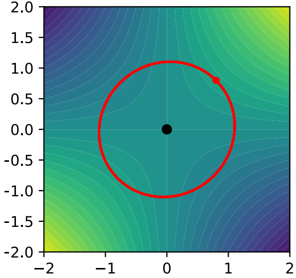
Remarks:
- ◦ GDA (i.e., $\mathbf{z}^{k+1}=\mathbf{z}^k-\tau\mathbf{V}(\mathbf{z}^k)$) diverges even for the simple bilinear objective (Lecture 11).
- ◦ Roughly speaking, minimax is harder than just optimization (Lecture 11).

A running, bilinear example: $\min_{x\in\mathbb{R}} \max_{y\in\mathbb{R}} xy$

- GDA



- Alternating GDA



GDA

1. Choose $\mathbf{x}^0, \mathbf{y}^0$ and τ.
2. For $k = 0, 1, \cdots$, perform:
   $\mathbf{x}^{k+1} := \mathbf{x}^k - \tau\nabla_{\mathbf{x}}\Phi(\mathbf{x}^k, \mathbf{y}^k)$.
   $\mathbf{y}^{k+1} := \mathbf{y}^k + \tau\nabla_{\mathbf{y}}\Phi(\mathbf{x}^k, \mathbf{y}^k)$.

AltGDA

1. Choose $\mathbf{x}^0, \mathbf{y}^0$ and α_k.
2. For $k = 0, 1, \cdots$, perform:
   $\mathbf{x}^{k+1} := \mathbf{x}^k - \tau\nabla_{\mathbf{x}}\Phi(\mathbf{x}^k, \mathbf{y}^k)$.
   $\mathbf{y}^{k+1} := \mathbf{y}^k + \tau\nabla_{\mathbf{y}}\Phi(\mathbf{x}^{k+1}, \mathbf{y}^k)$.

A preview of algorithms to be covered

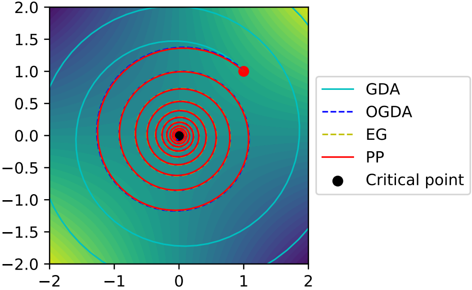


Figure: Trajectory of different algorithms for a simple bilinear game $\min_x \max_y xy$.

- Convergent algorithms in the sequel
  - Proximal point method (PPM)
  - Extra-gradient (EG)
  - Optimistic Gradient Descent Ascent (OGDA)

A preview of algorithms to be covered

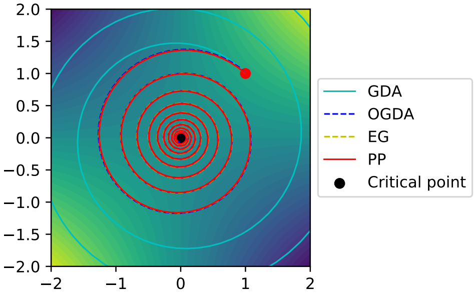


Figure: Trajectory of different algorithms for a simple bilinear game $\min_x \max_y xy$.

- Convergent algorithms in the sequel
  - Proximal point method (PPM)
  - Extra-gradient (EG)
  - Optimistic Gradient Descent Ascent (OGDA)
- EG and OGDA are approximations of the PPM [12]

Proximal point method (PPM)

◦ Consider following smooth unconstrained optimization problem: $\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x})$

Proximal point method for convex minimization.

For a step-size $\tau > 0$, PPM can be written as follows

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) + \frac{1}{2\tau}\|\mathbf{x} - \mathbf{x}^k\|^2 \right\} := \mathrm{prox}_{\tau f}(\mathbf{x}^k) \tag{3}$$

Observations: ◦ The optimality condition of (3) reveals a simpler PPM recursion for smooth f:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \tau\nabla f(\mathbf{x}^{k+1}).$$

◦ PPM is an **implicit**, non-practical algorithm since we need the point $\mathbf{x}^{k+1}$ for its update.

◦ Each step of PPM can be as hard as solving the original problem.

◦ Convergence properties are well understood due to Rockafellar [18].

PPM and minimax optimization

PPM applied to the minimax template: $\min_{\mathbf{x}\in\mathbb{R}^d}\max_{\mathbf{y}\in\mathbb{R}^n}\Phi(\mathbf{x},\mathbf{y})$

Define $\mathbf{z} = [\mathbf{x},\mathbf{y}]^\top$ and $\mathbf{V}(\mathbf{z}) = [\nabla_{\mathbf{x}}\Phi(\mathbf{x},\mathbf{y}), -\nabla_{\mathbf{y}}\Phi(\mathbf{x},\mathbf{y})]^\top$. PPM iterations with a step-size $\tau > 0$ is given by

$$\mathbf{z}^{k+1} = \mathbf{z}^k - \tau\mathbf{V}(\mathbf{z}^{k+1}).$$

Derivation:

- For $\tau > 0$, $(\mathbf{x}^{k+1},\mathbf{y}^{k+1})$ is the unique solution to the saddle point problem,

$$\min_{\mathbf{x}\in\mathbb{R}^d}\max_{\mathbf{y}\in\mathbb{R}^n}\Phi(\mathbf{x},\mathbf{y}) + \frac{1}{2\tau}\|\mathbf{x}-\mathbf{x}^k\|^2 - \frac{1}{2\tau}\|\mathbf{y}-\mathbf{y}^k\|^2 \tag{4}$$

- Writing the optimality condition of the update in (4)

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \tau\nabla_{\mathbf{x}}\Phi(\mathbf{x}^{k+1},\mathbf{y}^{k+1}), \qquad \mathbf{y}^{k+1} = \mathbf{y}^k + \tau\nabla_{\mathbf{y}}\Phi(\mathbf{x}^{k+1},\mathbf{y}^{k+1}) \tag{5}$$

Observation:

- **PPM is an implicit algorithm.**
- For the bilinear problem, PPM is implementable!

PPM guarantees for minimax optimization

Theorem (Convergence of PPM [18])

Suppose $(\mathbf{x}^k, \mathbf{y}^k)$ *be the iterates generated by PPM (i.e., (5)), then for the averaged iterates, it holds that*

$$\left| \Phi\left(\frac{1}{K}\sum_{k=1}^{K} \mathbf{x}^k, \frac{1}{K}\sum_{k=1}^{K} \mathbf{y}^k \right) - \Phi(\mathbf{x}^\star, \mathbf{y}^\star) \right| \le \frac{\|\mathbf{x}^0 - \mathbf{x}^\star\|^2 + \|\mathbf{y}^0 - \mathbf{y}^\star\|^2}{\tau K}.$$

Theorem (Linear convergence [18])

Suppose $(\mathbf{x}^k, \mathbf{y}^k)$ *be the iterates generated by* (5), $\Phi(\cdot,\cdot)$ *is* μ_x*−strongly convex in* $\mathbf{x}$ *and* μ_y*−strongly concave in* $\mathbf{y}$*. Let* $\mu = \max\{\mu_x, \mu_y\}$*. Then, for any* $\tau > 0$, $(\mathbf{x}^k, \mathbf{y}^k)$ *satisfies the following*

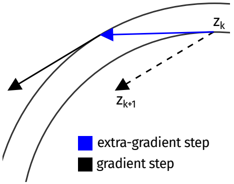
$$r^{k+1} \le \frac{1}{1+\mu\tau} r^k,$$

where $r^k = \|\mathbf{x}^k - \mathbf{x}^\star\|^2 + \|\mathbf{y}^k - \mathbf{y}^\star\|^2$.

Remark:
- ○ Still need an implementable and convergent algorithm beyond the stylized bilinear case.
- ○ Note what happens when $\tau \to \infty$.

Extra-gradient algorithm (EG) [8]

EG method for saddle point problems
1. Choose $\mathbf{x}^0, \mathbf{y}^0$ and τ.
2. For $k = 0, 1, \cdots,$ perform:
$\tilde{\mathbf{x}}^k := \mathbf{x}^k - \tau \nabla_{\mathbf{x}} \Phi(\mathbf{x}^k, \mathbf{y}^k),$
$\tilde{\mathbf{y}}^k := \mathbf{y}^k + \tau \nabla_{\mathbf{y}} \Phi(\mathbf{x}^k, \mathbf{y}^k).$
$\mathbf{x}^{k+1} := \mathbf{x}^k - \tau \nabla_{\mathbf{x}} \Phi(\tilde{\mathbf{x}}^k, \tilde{\mathbf{y}}^k).$
$\mathbf{y}^{k+1} := \mathbf{y}^k + \tau \nabla_{\mathbf{y}} \Phi(\tilde{\mathbf{x}}^k, \tilde{\mathbf{y}}^k).$



◦ Idea: Predict the gradient at the next point

$$\mathbf{z}^{k+1} = \mathbf{z}^k - \tau \mathbf{V}(\underbrace{\mathbf{z}^k - \tau \mathbf{V}(\mathbf{z}^k)}_{\text{prediction of } \mathbf{z}^{k+1}}) \tag{EG}$$

Remark: ◦ 1-extra-gradient computation per iteration

Extra-gradient algorithm: Convergence

Theorem (General case [12])

Let $0 < \tau \leq \frac{1}{L}$. *It holds that*

- *Iterates* $(\mathbf{x}^k, \mathbf{y}^k)$ *remains bounded in a convex compact set.*
- *Primal-dual gap reduces:* $\mathrm{Gap}\left(\frac{1}{K}\sum_{k=1}^{K}\mathbf{x}^k, \frac{1}{K}\sum_{k=1}^{K}\mathbf{y}^k\right) \leq \mathcal{O}\left(\frac{1}{K}\right)$.

Theorem (Linear convergence [13])

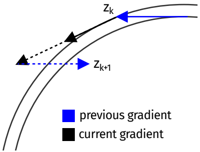
Suppose $(\mathbf{x}^k, \mathbf{y}^k)$ *be the iterates generated by Extra-gradient algorithm,* $\Phi(\cdot,\cdot)$ *is* μ_x*−strongly convex in* $\mathbf{x}$ *and* μ_y*−strongly concave in* $\mathbf{y}$. *Let* $\mu = \max\{\mu_x, \mu_y\}$. *Then, for* $\tau = \frac{1}{4L}$, $(\mathbf{x}^k, \mathbf{y}^k)$ *satisfies,*

$$r^{k+1} \leq \left(1 - \frac{1}{c\kappa}\right)^k r^0,$$

where $r^k = \|\mathbf{x}^k - \mathbf{x}^\star\|^2 + \|\mathbf{y}^k - \mathbf{y}^\star\|^2$, $\kappa = \frac{L}{\mu}$ *is the condition number of the problem, and* c *is a constant which is independent of the problem parameters.*

Optimistic gradient descent algorithm (OGDA) [17]

OGDA for saddle point problems
1. Choose $\mathbf{x}^0, \mathbf{y}^0, \mathbf{x}^1, \mathbf{y}^1$ and τ.
2. For $k = 1, \cdots,$ perform:
$\mathbf{x}^{k+1} := \mathbf{x}^k - 2\tau\nabla_{\mathbf{x}}\Phi(\mathbf{x}^k, \mathbf{y}^k) + \tau\nabla_{\mathbf{x}}\Phi(\mathbf{x}^{k-1}, \mathbf{y}^{k-1})$.
$\mathbf{y}^{k+1} := \mathbf{y}^k + 2\tau\nabla_{\mathbf{y}}\Phi(\mathbf{x}^k, \mathbf{y}^k) - \tau\nabla_{\mathbf{y}}\Phi(\mathbf{x}^{k-1}, \mathbf{y}^{k-1})$.



◦ Main difference from the GDA: Add a "momentum" or "reflection" term to the updates

$$\mathbf{z}^{k+1} = \mathbf{z}^k - \tau \left[\mathbf{V}(\mathbf{z}^k) + \underbrace{(\mathbf{V}(\mathbf{z}^k) - \mathbf{V}(\mathbf{z}^{k-1}))}_{\text{momentum}} \right]. \tag{OGDA}$$

◦ Known as Popov's method [16], it is also a special case of the Forward-Reflected-Backward method [11].

◦ It has ties to the Reflected-Forward-Backward Splitting (RFBS) method [3]:

$$\mathbf{z}^{k+1} = \mathbf{z}^k - \tau\mathbf{V}(2\mathbf{z}^k - \mathbf{z}^{k-1}). \tag{RFBS}$$

Remark: ◦ Advanced material at the end: OGDA is an approximation of PPM for bilinear problems.

OGDA: Convergence

Theorem (General case [12])

Let $0 < \tau \leq \frac{1}{2L}$, $\mathbf{x}^1 = \mathbf{x}^0, \mathbf{y}^1 = y^0$. *It holds that*

- *Iterates* $(\mathbf{x}^k, \mathbf{y}^k)$ *remains bounded in a convex compact set.*
- *Primal-dual gap reduces:* $\mathrm{Gap}\left(\frac{1}{K}\sum_{k=1}^{K}\mathbf{x}^k, \frac{1}{K}\sum_{k=1}^{K}\mathbf{y}^k\right) \leq \mathcal{O}\left(\frac{1}{K}\right)$.

Theorem (Linear convergence [13])

Suppose $(\mathbf{x}^k, \mathbf{y}^k)$ *be the iterates generated by OGDA,* $\Phi(\cdot,\cdot)$ *is* μ_x-*strongly convex in* $\mathbf{x}$ *and* μ_y-*strongly concave in* $\mathbf{y}$. *Let* $\mu = \max\{\mu_x, \mu_y\}$. *Then, for* $\tau = \frac{1}{4L}$, $(\mathbf{x}^k, \mathbf{y}^k)$ *satisfies,*

$$r^{k+1} \leq \left(1 - \frac{1}{c\kappa}\right)^k r^0,$$

where $r^k = \|\mathbf{x}^k - \mathbf{x}^\star\|^2 + \|\mathbf{y}^k - \mathbf{y}^\star\|^2$, $\kappa = \frac{L}{\mu}$ *is the condition number of the problem, and* c *is a constant which is independent of the problem parameters.*

A generalization of EG: The Mirror-Prox Algorithm

Definition: Bregman distance

Let $\omega : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ be a distance generating function where ω is $1-$strongly convex w.r.t. some norm $\|\cdot\|$ on the underlying space and is continuously differentiable. The Bregman distance induced by $\omega(\cdot)$ is given by

$$D_\omega(\mathbf{z}, \mathbf{z}') = \omega(\mathbf{z}) + \omega(\mathbf{z}') - \nabla\omega(\mathbf{z}')^\top(\mathbf{z} - \mathbf{z}').$$

Mirror-Prox algorithm

1. Choose $\mathbf{x}^0, \mathbf{y}^0$ and τ.
2. For $k = 0, 1, \cdots,$ perform:

$$\tilde{\mathbf{z}}^k = \arg\min_{\mathbf{z}\in\mathcal{X}\times\mathcal{Y}}(D_\omega(\mathbf{z}, \mathbf{z}^k) - \langle \tau\mathbf{V}(\mathbf{z}^k), \mathbf{z}\rangle).$$
$$\mathbf{z}^{k+1} = \arg\min_{\mathbf{z}\in\mathcal{X}\times\mathcal{Y}}(D_\omega(\mathbf{z}, \tilde{\mathbf{z}}^k) - \langle \tau\mathbf{V}(\tilde{\mathbf{z}}^k), \mathbf{z}\rangle).$$

Theorem (Mirror-Prox convergence)

Denote by $\Omega := \max_{\mathbf{z}\in\mathcal{X}\times\mathcal{Y}} D_\omega(\mathbf{z}, \mathbf{z}')$. The mirror-prox algorithm with $\tau \leq \frac{1}{L}$,

$$\mathrm{Gap}\left(\frac{1}{K}\sum_{k=1}^{K}\mathbf{x}^k, \frac{1}{K}\sum_{k=1}^{K}\mathbf{y}^k\right) \leq \mathcal{O}\left(\frac{\Omega}{K}\right).$$

Comparison of convergence rates for smooth convex-concave minimax

Method	Assumption on $\Phi(\cdot,\cdot)$	Convergence rate	Reference	Note
PP	convex-concave	$\mathcal{O}\left(\epsilon^{-1}\right)$	[18]	
PP	strongly convex- strongly concave	$\mathcal{O}\left(\kappa\log(\epsilon^{-1})\right)$	[18]	Implicit algorithm
PP	Bilinear	$\mathcal{O}\left(\kappa\log(\epsilon^{-1})\right)$	[18]	
EG	convex-concave	$\mathcal{O}\left(\epsilon^{-1}\right)$	[12]	
EG	strongly convex- strongly concave	$\mathcal{O}\left(\kappa\log(\epsilon^{-1})\right)$	[13, 12]	1 extra-gradient evaluation per iteration
EG	Bilinear	$\mathcal{O}\left(\kappa\log(\epsilon^{-1})\right)$	[13, 12]	
OGDA	convex-concave	$\mathcal{O}\left(\epsilon^{-1}\right)$	[12]	
OGDA	strongly convex- strongly concave	$\mathcal{O}\left(\kappa\log(\epsilon^{-1})\right)$	[13, 12]	no obvious downside
OGDA	Bilinear	$\mathcal{O}\left(\kappa\log(\epsilon^{-1})\right)$	[13, 12]	

Primal-dual methods for composite minimization: minimax reformulation

∘ Quest: Looking for algorithms such that $(\mathbf{x}^k, \mathbf{y}^k) \to (\mathbf{x}^\star, \mathbf{y}^\star)$ (with rates?)

Another restricted minimax template

$$\min_{\mathbf{x}\in\mathcal{X}} h(\mathbf{x}) + f(\mathbf{x}) + g(\mathbf{A}\mathbf{x}) = \min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y}) := h(\mathbf{x}) + f(\mathbf{x}) + \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle - g^*(\mathbf{y}).$$

We assume that

- $f(\mathbf{x}) : \mathcal{X} \to \mathbb{R}$ is proper, convex and lower-semicontinuous (l.s.c.),
- $h(\mathbf{x}) : \mathcal{X} \to \mathbb{R}$ is proper, convex, l.s.c. and differentiable with a $\frac{1}{\beta}$-Lipschitz continuous gradient,
- $g^*(\mathbf{y}) : \mathcal{Y} \to \mathbb{R}$ is proper, convex and l.s.c.
- $\mathcal{X} \subseteq R^p$ and $\mathcal{Y} \subseteq \mathbb{R}^n$,
- $\mathbf{A} : \mathcal{X} \to \mathcal{Y}$ is a bounded linear operator,
- Problem has at least one solution $(\mathbf{x}^\star, \mathbf{y}^\star) \in \mathcal{X} \times \mathcal{Y}$

Primal-dual hybrid gradient method (PDHG, aka Chambolle-Pock)

$$\min_{\mathbf{x}\in\mathcal{X}} h(\mathbf{x}) + f(\mathbf{x}) + g(\mathbf{A}\mathbf{x}) = \min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y}) := h(\mathbf{x}) + f(\mathbf{x}) + \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle - g^*(\mathbf{y})$$

PDHG [4], $(h(\mathbf{x}) = 0)$

1. Choose $\hat{\mathbf{x}}^0, \mathbf{x}^0, \mathbf{y}^0$ and $\tau, \sigma > 0$.

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

$$\mathbf{y}^{k+1} = \mathrm{prox}_{\sigma g^*}\left(\mathbf{y}^k + \sigma \mathbf{A}\tilde{\mathbf{x}}^k\right).$$

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\tau f}\left(\mathbf{x}^k - \tau \mathbf{A}^T\, \mathbf{y}^{k+1}\right).$$

$$\tilde{\mathbf{x}}^{k+1} = 2\mathbf{x}^{k+1} - \mathbf{x}^k.$$

Theorem ([4])

Let $L = \|A\|$, and choose τ and σ such that we have $\tau\sigma L^2 < 1$. Then, it holds that

- *Iterates $(\mathbf{x}^k, \mathbf{y}^k)$ remains bounded in a convex compact set.*
- *Primal-dual gap satisfies* $\mathrm{Gap}\left(\frac{1}{K}\sum_{k=1}^K \mathbf{x}^k, \frac{1}{K}\sum_{k=1}^K \mathbf{y}^k\right) \leq \mathcal{O}\left(\frac{1}{K}\right)$.
- $(\mathbf{x}^k, \mathbf{y}^k)$ *converges to saddle point* $(\mathbf{x}^\star, \mathbf{y}^\star)$.
- *If f and g are smooth, the rate improves to $\mathcal{O}(1/K^2)$.*
- *If f and g are also strongly convex, the convergence is linear.*

Stochastic PDHG

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) + \sum_{i=1}^{n} g_i(\mathbf{A}_i\mathbf{x}) = \min_{\mathbf{x}\in\mathcal{X}}\max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y}) := \underbrace{h(\mathbf{x})}_{=0} + f(\mathbf{x}) + \sum_{i=1}^{n}\langle \mathbf{A}_i\mathbf{x}, \mathbf{y}_i\rangle - \sum_{i=1}^{n} g_i^*(\mathbf{y}_i) \quad (6)$$

Algorithm 1 Stochastic Primal-Dual Hybrid Gradient

Input: Pick step sizes σ_i, τ and $\mathbf{x}^0 \in \mathcal{X}$, $\mathbf{y}^0 = \mathbf{y}^1 = \bar{\mathbf{y}}^1 \in \mathcal{Y}$. Given $\mathbf{P} = \mathrm{diag}(\mathsf{p}_1, \ldots, \mathsf{p}_n)$.

for $k = 1, 2, \ldots$ **do**

  $\mathbf{x}^k = \mathrm{prox}_{\tau f}(\mathbf{x}^{k-1} - \tau \sum_i \mathbf{A}_i^\top \bar{\mathbf{y}}_i^k)$

  Draw $j_k \in \{1, \ldots, n\}$ such that $\mathbb{P}(j_k = j) = \mathsf{p}_j$.

  $\mathbf{y}_{j_k}^{k+1} = \mathrm{prox}_{\sigma_{j_k} g_{j_k}^*}(\mathbf{y}_{j_k}^k + \sigma_{j_k}\mathbf{A}_{j_k}\mathbf{x}^k)$

  $\mathbf{y}_j^{k+1} = \mathbf{y}_j^k, \forall j \neq j_k$

  $\bar{\mathbf{y}}_i^{k+1} = \mathbf{y}_i^{k+1} + \mathbf{P}^{-1}(\mathbf{y}_i^{k+1} - \mathbf{y}_i^k), \forall i,$

end for

Remarks:

- Note: $\mathsf{p}_i^{-1}\tau\sigma_i\|A_i\|^2 < 1$.
- Only one dual vector is updated at each iteration.
- Especially effective when $\mathbf{A}_i$ is row-vector.

SPDHG: Convergence [1]

Theorem (Almost sure convergence)

Almost surely, there exists $(\mathbf{x}^\star, \mathbf{y}^\star) \in \mathcal{Z}^\star$*, such that the iterates of SPDHG satisfy* $\mathbf{x}^k \to \mathbf{x}^\star$ *and* $\mathbf{y}^k \to \mathbf{y}^\star$*.*

Theorem (Sublinear convergence)

Define the ergodic sequences $\mathbf{x}_{av}^K = \sum_{k=1}^K \mathbf{x}^k$ *and* $\mathbf{y}_{av}^{K+1} = \sum_{k=1}^K \mathbf{y}^{k+1}$*, and define the gap function*

$$\mathrm{Gap}(\mathbf{x}_{av}^K, \mathbf{y}_{av}^{K+1}) = \sup_{\mathbf{x},\mathbf{y}} f(\mathbf{x}_{av}^K) + \langle A\mathbf{x}_{av}^K, \mathbf{y}\rangle - g^*(\mathbf{y}) - f(\mathbf{x}) - \langle A\mathbf{x}, \mathbf{y}_{av}^{K+1}\rangle + g^*(\mathbf{y}_{av}^{K+1}).$$

The following result holds for the expected primal-dual gap, which is expectation of a supremum

$$\mathbb{E}\left[\mathrm{Gap}(\mathbf{x}_{av}^K, \mathbf{y}_{av}^{K+1})\right] = \mathcal{O}\left(\frac{1}{K}\right). \tag{7}$$

Primal-dual algorithms for minimax: The zoo!

$$\min_{\mathbf{x}\in\mathcal{X}} h(\mathbf{x}) + f(\mathbf{x}) + g(\mathbf{A}\mathbf{x}) = \min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y}) := h(\mathbf{x}) + f(\mathbf{x}) + \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle - g^*(\mathbf{y})$$

3 operator splitting [6], ($\mathbf{A} = \mathbb{I}$)

1. Choose $\hat{\mathbf{x}}^0, \mathbf{x}^0, \mathbf{y}^0$ and $\tau > 0$.

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

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\tau f}\left(\tilde{\mathbf{x}}^k\right).$$
$$\mathbf{y}^{k+1} = \frac{1}{\tau}(\mathbb{I} + \mathrm{prox}_{\tau^{-1} g})\left(2\mathbf{x}^{k+1} - \tilde{\mathbf{x}}^k - \tau\nabla h(\mathbf{x}^{k+1})\right).$$
$$\tilde{\mathbf{x}}^{k+1} = \mathbf{x}^{k+1} - \tau\nabla h(\mathbf{x}^{k+1}) - \tau\mathbf{y}^{k+1}.$$

Primal-dual algorithms for minimax: The zoo!

$$\min_{\mathbf{x}\in\mathcal{X}} h(\mathbf{x}) + f(\mathbf{x}) + g(\mathbf{A}\mathbf{x}) = \min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y}) := h(\mathbf{x}) + f(\mathbf{x}) + \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle - g^*(\mathbf{y})$$

3 operator splitting [6], ($\mathbf{A} = \mathbb{I}$)

1. Choose $\hat{\mathbf{x}}^0, \mathbf{x}^0, \mathbf{y}^0$ and $\tau > 0$.
2. For $k = 0, 1, \cdots$, perform:

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\tau f}\left(\tilde{\mathbf{x}}^k\right).$$

$$\mathbf{y}^{k+1} = \frac{1}{\tau}(\mathbb{I} + \mathrm{prox}_{\tau^{-1} g})\left(2\mathbf{x}^{k+1} - \tilde{\mathbf{x}}^k - \tau\nabla h(\mathbf{x}^{k+1})\right).$$

$$\tilde{\mathbf{x}}^{k+1} = \mathbf{x}^{k+1} - \tau\nabla h(\mathbf{x}^{k+1}) - \tau\mathbf{y}^{k+1}.$$

- There is a stochastic variant [24].

Primal-dual algorithms for minimax: The zoo!

$$\min_{\mathbf{x}\in\mathcal{X}} h(\mathbf{x}) + f(\mathbf{x}) + g(\mathbf{A}\mathbf{x}) = \min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y}) := h(\mathbf{x}) + f(\mathbf{x}) + \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle - g^*(\mathbf{y})$$

Condat-Vu [5, 21]

1. Choose $\hat{\mathbf{x}}^0, \mathbf{x}^0, \mathbf{y}^0$ and $\tau, \sigma > 0$.
2. For $k = 0, 1, \cdots,$ perform:

$$\mathbf{y}^{k+1} = \mathrm{prox}_{\sigma g^*}\left(\mathbf{y}^k + \sigma\mathbf{A}\tilde{\mathbf{x}}^k\right).$$

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\tau f}\left(\mathbf{x}^k - \tau\nabla h(\mathbf{x}^k) - \tau\mathbf{A}^T\ \mathbf{y}^{k+1}\right).$$

$$\tilde{\mathbf{x}}^{k+1} = 2\mathbf{x}^{k+1} - \mathbf{x}^k.$$

Primal-dual algorithms for minimax: The zoo!

$$\min_{\mathbf{x}\in\mathcal{X}} h(\mathbf{x}) + f(\mathbf{x}) + g(\mathbf{A}\mathbf{x}) = \min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y}) := h(\mathbf{x}) + f(\mathbf{x}) + \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle - g^*(\mathbf{y})$$

PD3O splitting [23]

1. Choose $\hat{\mathbf{x}}^0, \mathbf{x}^0, \mathbf{y}^0$ and $\tau, \sigma > 0$.
2. For $k = 0, 1, \cdots,$ perform:

$$\mathbf{y}^{k+1} = \mathrm{prox}_{\sigma g^*}\left(\mathbf{y}^k + \sigma \mathbf{A}\tilde{\mathbf{x}}^k\right).$$

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\tau f}\left(\mathbf{x}^k - \tau\nabla h(\mathbf{x}^k) - \tau\mathbf{A}^T\,\mathbf{y}^{k+1}\right).$$

$$\tilde{\mathbf{x}}^{k+1} = 2\mathbf{x}^{k+1} - \mathbf{x}^k + \tau\nabla h(\mathbf{x}^k) - \tau\nabla h(\mathbf{x}^{k+1}).$$

Between convex-concave and nonconvex-nonconcave

Nonconvex-concave problems

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y})$$

◦ $\Phi(\mathbf{x},\mathbf{y})$ is nonconvex in $\mathbf{x}$, concave in $\mathbf{y}$, smooth in $\mathbf{x}$ and $\mathbf{y}$.

Recall

Define $f(\mathbf{x}) = \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y})$.

◦ Gradient descent applied to nonconvex f requires $\mathcal{O}(\epsilon^{-2})$ iterations to give an ϵ-stationary point.

◦ (Sub)gradient of f can be computed using Danskin's theorem:

$$\nabla_{\mathbf{x}}\Phi(\cdot, y^\star(\cdot)) \in \partial f(\cdot), \text{ where } y^\star(\cdot) \in \arg\max_{\mathbf{y}\in\mathcal{Y}} \Phi(\cdot,\mathbf{y}),$$

which is tractable since Φ is concave in $\mathbf{y}$ [9].

Remark: ◦ "Conceptually" much easier than nonconvex-nonconcave case.

A summary of results for nonconvex-concave setting

◦ A summary of gradient complexities to reach $\epsilon-$first order stationary point in terms of **gradient mapping.**

Method	Assumption on $\Phi(\cdot,\cdot)$	Convergence rate	Reference
GDA	**noconvex-concave**	$\tilde{\mathcal{O}}\left(\epsilon^{-6}\right)$	**[9]**
GDA	**nonconvex- strongly concave**	$\mathcal{O}\left(\epsilon^{-2}\right)$	**[9]**
GDmax	**nonconvex-concave**	$\tilde{\mathcal{O}}\left(\epsilon^{-6}\right)$	**[7]**
GDmax	**nonconvex- strongly concave**	$\mathcal{O}\left(\epsilon^{-2}\right)$	**[7]**
HiBSA, AGP	**nonconvex-concave**	$\tilde{\mathcal{O}}\left(\epsilon^{-4}\right)$	**[10], [22]**
HiBSA, AGP	**nonconvex- strongly concave**	$\mathcal{O}\left(\epsilon^{-2}\right)$	**[10], [22]**

The elephant in the room: Nonsmooth, nonconvex optimization

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

- Finding a stationary point of nonsmooth nonconvex minimization problems are hard [25]
  - A traditional $\epsilon-$stationarity can not be obtained in finite time
- Even the relax notions are hard [19]
- Really puzzling how deep learning approaches with ReLu etc. work...

How about purely primal approaches?

$$\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) : \mathbf{A}\mathbf{x} = \mathbf{b} \right\}$$

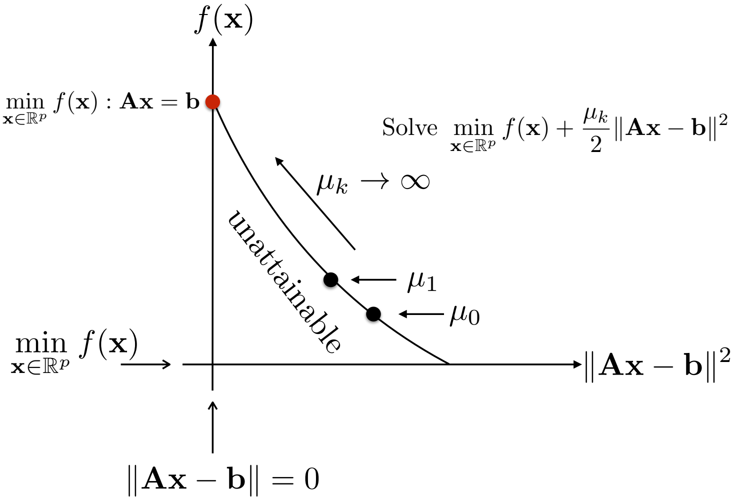
Penalty methods

- Convert constrained problem (difficult) to unconstrained (easy).
- Penalized function with penalty parameter $\mu > 0$:

$$F_\mu(\mathbf{x}) := \left\{ f(\mathbf{x}) + \frac{\mu}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 \right\} \quad \overset{\mu\to\infty}{\Longleftrightarrow} \quad \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) : \mathbf{A}\mathbf{x} = \mathbf{b} \right\}.$$

- **Observations**:
  - Minimize a weighted combination of $f(\mathbf{x})$ and $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$ at the same time.
  - μ determines the weight of $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$.
  - As $\mu \to \infty$, we enforce $\mathbf{A}\mathbf{x} = \mathbf{b}$.
  - Other functions than the quadratic $\frac{1}{2}\|\cdot\|^2$ are also possible e.g., exact nonsmooth penalty functions:
    - $\mu\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$ or $\mu\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_1$
    - They work with finite μ, but they are difficult to solve [15, Section 17.2], [2]

Quadratic penalty: Intuition



Quadratic penalty: Conceptual algorithm

Quadratic penalty method (QP):
1. Choose $\mathbf{x}_0 \in \mathbb{R}^p$ and $\mu_0 > 0$.
2. For $k = 0, 1, \cdots,$ perform:
2.a. $\mathbf{x}_k := \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) + \frac{\mu_k}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 \right\}.$
2.b. Update $\mu_{k+1} > \mu_k$.

Theorem [15, Theorem 17.1]

Assume that f is smooth and $\mu_k \to \infty$. Then, every limit point $\bar{\mathbf{x}}$ of the sequence $\{\mathbf{x}_k\}$ is a solution of the constrained problem

$$\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) \colon \mathbf{A}\mathbf{x} = \mathbf{b} \right\}.$$

Limitation

- The minimization problems of step 2.a. of the algorithm become ill-conditioned as $\mu_k \to \infty$.
- Common improvements:
  - Solve the subproblem inexactly, *i.e.*, up to ϵ accuracy.
  - Linearization to simplify subproblems (up next).

Quadratic penalty: Linearization

Generalized quadratic penalty method:
1. Choose $\mathbf{x}_0 \in \mathbb{R}^p$, $\mu_0 > 0$ and positive semidefinite matrix $\mathbf{Q}_k$.
2. For $k = 0, 1, \cdots,$ perform:
2.a. $\mathbf{x}_k := \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) + \frac{\mu_k}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2 + \frac{1}{2}\|\mathbf{x}-\mathbf{x}_{k-1}\|^2_{\mathbf{Q}_k} \right\}$.
2.b. Update $\mu_{k+1} > \mu_k$.

Ideas

- Minimize a majorizer of $F_\mu(\mathbf{x})$, parametrized by $\mathbf{Q}_k$ in step 2.a..
- $\mathbf{Q}_k = \mathbf{0}$ gives the standard QP; $\mathbf{Q}_k = \mathbf{I}$ gives strongly convex subproblems.
- $\mathbf{Q}_k = \alpha_k \mathbf{I} - \mu_k \mathbf{A}^\top \mathbf{A}$, with $\alpha_k \geq \mu_k \|\mathbf{A}\|^2$ gives

$$\mathbf{x}_k = \mathrm{prox}_{\frac{1}{\alpha_k} f}\left(\mathbf{x}_{k-1} - \frac{\mu_k}{\alpha_k}\mathbf{A}^\top(\mathbf{A}\mathbf{x}_{k-1} - \mathbf{b})\right)$$

Only one proximal operator!

and picking $\alpha_k = \mu_k \|\mathbf{A}\|^2$ gives

$$\mathbf{x}_k = \mathrm{prox}_{\frac{1}{\mu_k \|\mathbf{A}\|^2} f}\left(\mathbf{x}_{k-1} - \frac{1}{\|\mathbf{A}\|^2}\mathbf{A}^\top(\mathbf{A}\mathbf{x}_{k-1} - \mathbf{b})\right).$$

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

Recall: Prox-operator

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

Key properties:

- single valued & non-expansive since f is a proper convex function.
- 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.

Quadratic penalty: Linearized methods

Linearized QP method (LQP)	Accelerated linearized QP method (ALQP)
1. Choose $\mathbf{x}_0 \in \mathbb{R}^p$, $\sigma_0 = 1$, $\mu_0 > 0$.	1. Choose $\mathbf{x}_0, \mathbf{y}_0 \in \mathbb{R}^p$, $\tau_0 = 1$, $\mu_0 > 0$.
2. For $k = 0, 1, \cdots$:	2. For $k = 0, 1, \cdots$:
2.a. $\mathbf{x}_{k+1} := \mathrm{prox}_{\frac{1}{\mu_k \|\mathbf{A}\|^2} f}\left(\mathbf{x}_k - \frac{1}{\|\mathbf{A}\|^2}\mathbf{A}^\top(\mathbf{A}\mathbf{x}_k - \mathbf{b})\right)$.	2.a. $\mathbf{x}_{k+1} := \mathrm{prox}_{\frac{1}{\mu_k \|\mathbf{A}\|^2} f}\left(\mathbf{y}_k - \frac{1}{\|\mathbf{A}\|^2}\mathbf{A}^\top(\mathbf{A}\mathbf{y}_k - \mathbf{b})\right)$.
2.b. Update σ_{k+1} s.t. $\frac{(1-\sigma_{k+1})^2}{\sigma_{k+1}} = \frac{1}{\sigma_k}$.	2.b. $\mathbf{y}_{k+1} := \mathbf{x}_{k+1} + \frac{\tau_{k+1}(1-\tau_k)}{\tau_k}(\mathbf{x}_{k+1} - \mathbf{x}_k)$.
2.c. Update $\mu_{k+1} = \sqrt{\sigma_{k+1}}$.	2.c. Update $\mu_{k+1} = \mu_k(1 + \tau_{k+1})$.
	2.d. Update $\tau_{k+1} \in (0, 1)$ as the unique positive root of $\tau^3 + \tau^2 + \tau_k^2\tau - \tau_k^2 = 0$.

Theorem (Convergence [20])

- ***LQP***:

$$
\begin{aligned}
|f(\mathbf{x}_k) - f(\mathbf{x}^\star)| &\leq \mathcal{O}\left(\mu_0 k^{-1/2} + \mu_0^{-1} k^{-1/2}\right) \\
\|\mathbf{A}\mathbf{x}_k - \mathbf{b}\| &\leq \mathcal{O}\left(\mu_0^{-1} k^{-1/2}\right)
\end{aligned}
$$

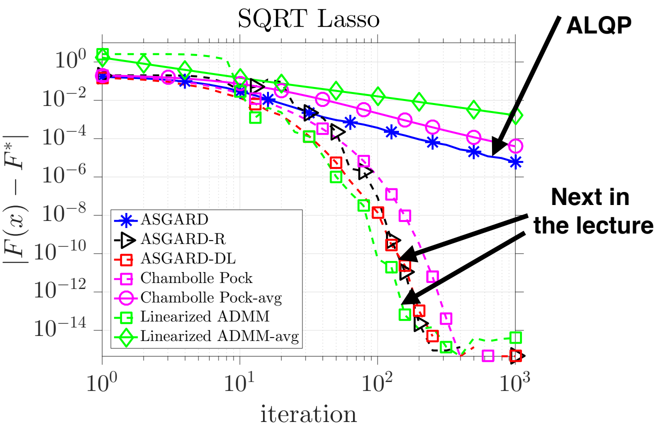
- ***ALQP***:

$$
\begin{aligned}
|f(\mathbf{x}_k) - f(\mathbf{x}^\star)| &\leq \mathcal{O}\left(\mu_0 k^{-1} + \mu_0^{-1} k^{-1}\right) \\
\|\mathbf{A}\mathbf{x}_k - \mathbf{b}\| &\leq \mathcal{O}\left(\mu_0^{-1} k^{-1}\right)
\end{aligned}
$$

In practice: poor (worst case) performance

◦ A nonsmooth problem: SQRT Lasso

$$\min_{\mathbf{x}\in\mathbb{R}^p} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2 + \lambda\|\mathbf{x}\|_1.$$



Wrap up!

◦ Recitation continues with Homework #2 on Friday...

*OGDA as an approximation of PPM

Claim: OGDA is an approximation of PPM.

◦ Consider the bilinear case $\Phi(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{B}\mathbf{y} \rangle$, where $\mathbf{B} \in \mathbb{R}^{p \times p}$ is a square full rank matrix. The point $(\mathbf{x}^\star, \mathbf{y}^\star) = (\mathbf{0}, \mathbf{0})$ is a unique saddle point.

◦ OGDA updates are

$$\mathbf{x}^{k+1} = \mathbf{x}^k - 2\tau\mathbf{B}\mathbf{y}^k + \tau\mathbf{B}\mathbf{y}^{k-1}, \qquad \mathbf{y}^{k+1} = \mathbf{y}^k + 2\tau\mathbf{B}^\top\mathbf{x}^k - \tau\mathbf{B}^\top\mathbf{x}^{k-1}$$

◦ From (5) , PP update on the variable $\mathbf{x}$ is

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \tau\mathbf{B}\mathbf{y}^{k+1} = \mathbf{x}^k - \tau\mathbf{B}\left(\mathbf{y}^k + \tau\mathbf{B}^\top\mathbf{x}^{k+1}\right),$$

where we used $\mathbf{y}^{k+1} = \mathbf{y}^k + \tau\mathbf{B}^\top\mathbf{x}^{k+1}$. So, PP method update on the variable $\mathbf{x}$ can be rewritten as

$$\mathbf{x}^{k+1} = (\mathbb{I} + \tau^2\mathbf{B}\mathbf{B}^\top)^{-1}(\mathbf{x}^k - \tau\mathbf{B}\mathbf{y}^k)$$

◦ Use the fact that $(\mathbb{I} - \tau^2\mathbf{B}\mathbf{B}^\top)$ is an approximation $(\mathbb{I} + \tau^2\mathbf{B}\mathbf{B}^\top)^{-1}$ with an error $o(\tau^2)$.

$$\left(\mathbb{I} + \tau^2\mathbf{B}\mathbf{B}^\top\right)^{-1} = \left(\mathbb{I} - \tau^2\mathbf{B}\mathbf{B}^\top + o(\tau^2)\right) \tag{8}$$

*OGDA as an approximation of PPM

- Using (8), rewrite the update on $\mathbf{x}$ for PPM as

$$\mathbf{x}^{k+1} = \left(\mathbb{I} - \tau^2 \mathbf{B}\mathbf{B}^\top + o(\tau^2)\right)(\mathbf{x}^k - \tau \mathbf{B}\mathbf{y}^k)$$

- Adding and subtracting $\mathbf{B}\mathbf{y}^k$ to the right hand side, using the PP updates and reorganizing the terms

$$\begin{aligned}
\mathbf{x}^{k+1} &= \mathbf{x}^k - \tau \mathbf{B}\mathbf{y}^k - \tau \mathbf{B}\left(\tau \mathbf{B}^\top \mathbf{x}^k - \tau^2 \mathbf{B}^\top \mathbf{B}\mathbf{y}^k\right) + o(\tau^2) \\
&= \mathbf{x}^k - 2\tau \mathbf{B}\mathbf{y}^k - \tau \mathbf{B}\left(\tau \mathbf{B}^\top \mathbf{x}^k - (\mathbb{I} + \tau^2 \mathbf{B}^\top \mathbf{B})\mathbf{y}^k\right) + o(\tau^2) \\
&= \mathbf{x}^k - 2\tau \mathbf{B}\mathbf{y}^k - \tau \mathbf{B}\left(\tau \mathbf{B}^\top \mathbf{x}^k - \mathbf{y}^{k-1} - \tau \mathbf{B}^\top \mathbf{x}^{k-1}\right) + o(\tau^2) \\
&= \mathbf{x}^k - 2\tau \mathbf{B}\mathbf{y}^k - \tau \mathbf{B}\mathbf{y}^{k-1} + o(\tau^2)
\end{aligned}$$

- The last equation is OGDA update for variable $\mathbf{x}$ plus an additional error of $o(\tau^2)$. Similarly for variable $\mathbf{y}$.

*OGDA as an approximation of PPM

- Using (8), rewrite the update on $\mathbf{x}$ for PPM as

$$\mathbf{x}^{k+1} = \left(\mathbb{I} - \tau^2 \mathbf{B}\mathbf{B}^\top + o(\tau^2)\right)(\mathbf{x}^k - \tau \mathbf{B}\mathbf{y}^k)$$

- Adding and subtracting $\mathbf{B}\mathbf{y}^k$ to the right hand side, using the PP updates and reorganizing the terms

$$\begin{aligned}
\mathbf{x}^{k+1} &= \mathbf{x}^k - \tau \mathbf{B}\mathbf{y}^k - \tau \mathbf{B}\left(\tau \mathbf{B}^\top \mathbf{x}^k - \tau^2 \mathbf{B}^\top \mathbf{B}\mathbf{y}^k\right) + o(\tau^2) \\
&= \mathbf{x}^k - 2\tau \mathbf{B}\mathbf{y}^k - \tau \mathbf{B}\left(\tau \mathbf{B}^\top \mathbf{x}^k - (\mathbb{I} + \tau^2 \mathbf{B}^\top \mathbf{B})\mathbf{y}^k\right) + o(\tau^2) \\
&= \mathbf{x}^k - 2\tau \mathbf{B}\mathbf{y}^k - \tau \mathbf{B}\left(\tau \mathbf{B}^\top \mathbf{x}^k - \mathbf{y}^{k-1} - \tau \mathbf{B}^\top \mathbf{x}^{k-1}\right) + o(\tau^2) \\
&= \mathbf{x}^k - 2\tau \mathbf{B}\mathbf{y}^k - \tau \mathbf{B}\mathbf{y}^{k-1} + o(\tau^2)
\end{aligned}$$

- The last equation is OGDA update for variable $\mathbf{x}$ plus an additional error of $o(\tau^2)$. Similarly for variable $\mathbf{y}$.

Proposition

Given a point $(\mathbf{x}^k, \mathbf{y}^k)$, let $(\hat{\mathbf{x}}^{k+1}, \hat{\mathbf{y}}^{k+1})$ be the point generated by performing a PP update on $(\mathbf{x}^k, \mathbf{y}^k)$, and let $(\mathbf{x}^{k+1}, \mathbf{y}^{k+1})$ be the point generated by performing an OGDA update on $(\mathbf{x}^k, \mathbf{y}^k)$. For $\eta > 0$

$$\|\mathbf{x}^{k+1} - \hat{\mathbf{x}}^{k+1}\| \leq o(\tau^2), \qquad \|\mathbf{y}^{k+1} - \hat{\mathbf{y}}^{k+1}\| \leq o(\tau^2).$$

*Tools for the algorithms: resolvent operator and prox-mapping

∘ We need to solve problems of type (9) at each iteration.

$$\boxed{\mathbf{x}^+ = \arg\min_{\mathbf{x}} \left\{ f(\mathbf{x}) + \frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\tau} \right\}} := \mathrm{prox}_{\tau f}(\mathbf{y}) \tag{9}$$

∘ Writing the optimality condition gives

$$0 \in \partial f(\mathbf{x}^+) + \frac{1}{\tau}(\mathbf{x}^+ - \mathbf{y}) \quad \Rightarrow \quad \mathbf{x}^+ = \underbrace{(\mathbb{I} + \tau\partial f)^{-1}}_{\text{resolvent operator}}(\mathbf{y}), \tag{10}$$

where ∂f is the subgradient of convex function f and $\mathbb{I}$ is the identity operator.

∘ We assume resolvent operator defined through (10) is either
- have a closed form solution, or
- can be efficiently solved.

⋆Tools for the algorithms: Moreau's identity

∘ Similarly, for the dual parameter update, we need the following proximal operator of g^*.

$$\mathbf{y}^+ = \mathrm{prox}_{\sigma g^*}(\mathbf{x})$$

∘ A fundamental equality for the prox operator: Moreau's identity

$$\mathbf{x} = \mathrm{prox}_g(\mathbf{x}) + \mathrm{prox}_{g^*}(\mathbf{x}) \qquad \text{(Moreau's Identity)}$$

∘ It is easy to compute $\mathrm{prox}_{\sigma g^*}(\mathbf{x})$ by using the proximal mapping of function g as

$$\mathrm{prox}_{\sigma g^*}(\mathbf{x}) = \mathbf{x} - \sigma\, \mathrm{prox}_{\sigma^{-1} g}\left(\frac{\mathbf{x}}{\sigma}\right) \qquad \text{(Extended Moreau's Identity)}$$

*Extended Moreau's identity

$$\boxed{\mathrm{prox}_{\sigma g^*}(\mathbf{x}) = \mathbf{x} - \sigma\, \mathrm{prox}_{\sigma^{-1} g}\left(\frac{\mathbf{x}}{\sigma}\right)}$$

Proof: Extended Moreau's identity

First prove that Moreau's identity holds: $\mathbf{x} = \mathrm{prox}_g(\mathbf{x}) + \mathrm{prox}_{g^*}(\mathbf{x})$

$$\mathbf{y} = \mathrm{prox}_g(\mathbf{x}) \iff \mathbf{x} - \mathbf{y} \in \partial g(\mathbf{y}) \qquad \text{(Optimality of prox)}$$

$$\iff \mathbf{y} \in \partial g^*(\mathbf{x} - \mathbf{y}) \qquad \text{(Conjugate subgradient theorem)}$$

$$\iff \mathbf{x} - \mathbf{y} = \mathrm{prox}_{g^*}(\mathbf{x})$$

$$\iff \mathbf{x} = \mathrm{prox}_g(\mathbf{x}) + \mathrm{prox}_{g^*}(\mathbf{x}) \qquad (\mathbf{y} = \mathrm{prox}_g(\mathbf{x}))$$

Now applying Moreau's identity to function σg

$$\mathbf{x} = \mathrm{prox}_{\sigma g}(\mathbf{x}) + \mathrm{prox}_{(\sigma g)^*}(\mathbf{x})$$

$$= \mathrm{prox}_{\sigma g}(\mathbf{x}) + \sigma \mathrm{prox}_{\sigma^{-1} g^*}\left(\frac{\mathbf{x}}{\sigma}\right) \qquad \left((\sigma g)^*(\mathbf{y}) = \sigma g^*\left(\frac{\mathbf{x}}{\sigma}\right)\right)$$

*Primal-dual with random extrapolation and coordinate descent: PURE-CD

Input: $\mathbf{x}_0 \in \mathbb{R}^n$, $\mathbf{y}_0 \in \mathbb{R}^m$
Parameters: $\theta = \mathrm{diag}(\theta_1, \ldots, \theta_m)$ is chosen as $\theta_j = \frac{\pi_j}{\underline{p}}$, where $\pi_j = \sum_{i \in I(j)} p_i$, and $\underline{p} = \min_i p_i$, and

$\tau_i < \frac{2p_i - \underline{p}}{\beta_i p_i + \underline{p}^{-1} p_i \sum_{j=1}^{m} \pi_j \sigma_j A_{j,i}^2}$[1].

for $k \in \mathbb{N}$ **do**

$\bar{\mathbf{y}}_{k+1} = \mathrm{prox}_{\sigma g^*}(\mathbf{y}_k + \sigma \mathbf{A}\mathbf{x}_k)$

$\bar{\mathbf{x}}_{k+1} = \mathrm{prox}_{\tau f}(\mathbf{x}_k - \tau \nabla h(\mathbf{x}_k) - \tau \mathbf{A}^\top \bar{\mathbf{y}}_{k+1})$

Draw $i_{k+1} \in \{1, \ldots, n\}$ randomly w.p. $\mathbb{P}(i_{k+1} = i) = p_i$

$\mathbf{x}_{k+1}^{i_{k+1}} = \bar{\mathbf{x}}_{k+1}^{i_{k+1}}$

$\mathbf{x}_{k+1}^{j} = \mathbf{x}_{k}^{j}, \forall j \neq i_{k+1}$

$\mathbf{y}_{k+1}^{j} = \bar{\mathbf{y}}_{k+1}^{j} + \sigma_j \theta_j [\mathbf{A}(\mathbf{x}_{k+1} - \mathbf{x}_k)]_j, \forall j \in J(i_{k+1})$

$\mathbf{y}_{k+1}^{j} = \mathbf{y}_{k}^{j}, \forall j \notin J(i_{k+1})$

end for

step size w. dense $\mathbf{A}$	iter. cost
$n\tau_i\sigma\|\mathbf{A}_i\|^2 < 1$	$\mathrm{nnz}(\mathbf{A}_i)$

[1] β_i are coordinate-wise Lipschitz constants of ∇f

⋆Experiments

- Datasets with varying sparsity levels, sparse, moderately sparse, and dense.
- Comparison with dense friendly SPDHG (Chambolle et al., 2018), sparse friendly VC-CD (Fercoq&Bianchi, 2019) with duplication[2].
- PURE-CD stays efficient in all cases, attaining best of both worlds.

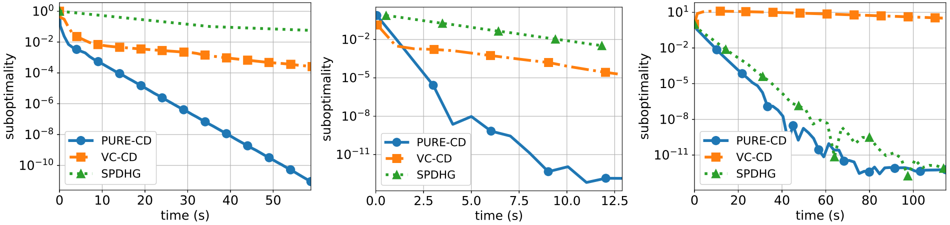


Figure: Lasso: Left: rcv1, $n = 20,242$, $m = 47,236$, density $= 0.16\%$, $\lambda = 10$; Middle: w8a, $n = 49,749$, $m = 300$, density $= 3.9\%$, $\lambda = 10^{-1}$; Right: covtype, $n = 581,012$, $m = 54$, density $= 22.1\%$, $\lambda = 10$.

[2] Fercoq, Bianchi, A coordinate-descent primal-dual algorithm with large step size and possibly nonseparable functions, SIOPT, 2019.

⋆Experiments

- Strongly convex strongly concave ridge regression problems with varying regularization parameter.
- PURE-CD is competitive with state-of-the-art specialized methods for this problem.

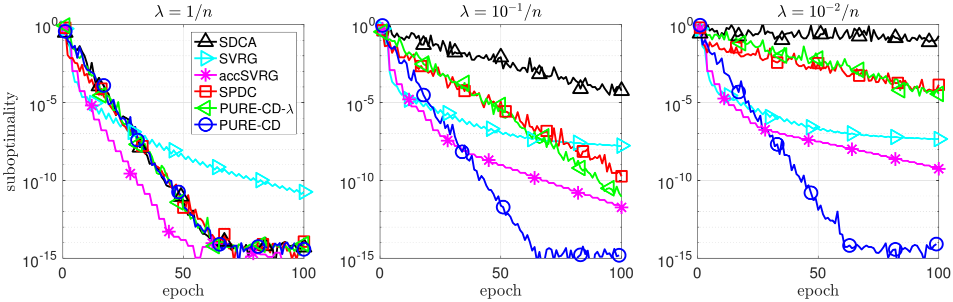


Figure: Ridge. a9a, $n = 32,561$, $m = 123$.

References I

[1] Ahmet Alacaoglu, Olivier Fercoq, and Volkan Cevher.
On the convergence of stochastic primal-dual hybrid gradient.
arXiv preprint arXiv:1911.00799, 2019.

[2] Dimitri P Bertsekas.
Necessary and sufficient conditions for a penalty method to be exact.
Mathematical programming, 9(1):87–99, 1975.

[3] Volkan Cevher and B?ng Công Vũ.
A reflected forward-backward splitting method for monotone inclusions involving lipschitzian operators.
Set-Valued and Variational Analysis, pages 1–12, 2020.

[4] 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.

[5] L. Condat.
A primal–dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms.
J. Optim. Theory Appl., 158:460–479, 2013.

References II

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

[7] Chi Jin, Praneeth Netrapalli, and Michael I Jordan.
What is local optimality in nonconvex-nonconcave minimax optimization?
arXiv preprint arXiv:1902.00618, 2019.

[8] G. M. Korpelevic.
An extragradient method for finding saddle-points and for other problems.
Èkonom. i Mat. Metody., 12(4):747–756, 1976.

[9] Tianyi Lin, Chi Jin, and Michael I Jordan.
On gradient descent ascent for nonconvex-concave minimax problems.
arXiv preprint arXiv:1906.00331, 2019.

[10] Songtao Lu, Ioannis Tsaknakis, Mingyi Hong, and Yongxin Chen.
Hybrid block successive approximation for one-sided non-convex min-max problems: algorithms and applications.
IEEE Transactions on Signal Processing, 2020.

References III

[11] Yura Malitsky and Matthew K Tam.
A forward-backward splitting method for monotone inclusions without cocoercivity.
SIAM Journal on Optimization, 30(2):1451–1472, 2020.

[12] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil.
Convergence rate of $\mathcal{O}(1/k)$ for optimistic gradient and extra-gradient methods in smooth convex-concave saddle point problems.
arXiv preprint arXiv:1906.01115, 2019.

[13] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil.
A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach.
In *International Conference on Artificial Intelligence and Statistics*, pages 1497–1507. PMLR, 2020.

[14] A. Nemirovskii.
Prox-method with rate of convergence $\mathcal{O}(1/t)$ for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems.
SIAM J. Op, 15(1):229–251, 2004.

[15] J. Nocedal and S.J. Wright.
Numerical Optimization.
Springer Series in Operations Research and Financial Engineering. Springer, 2 edition, 2006.

References IV

[16] Leonid Denisovich Popov.
A modification of the arrow-hurwicz method for search of saddle points.
Mathematical notes of the Academy of Sciences of the USSR, 28(5):845–848, 1980.

[17] Sasha Rakhlin and Karthik Sridharan.
Optimization, learning, and games with predictable sequences.
In *Advances in Neural Information Processing Systems*, pages 3066–3074, 2013.

[18] R. T. Rockafellar.
Augmented lagrangians and applications of the proximal point algorithm in convex programming.
Mathemathics of Operations Research, 1:97–116, 1976.

[19] Ohad Shamir.
Can we find near-approximately-stationary points of nonsmooth nonconvex functions?
arXiv preprint arXiv:2002.11962, 2020.

[20] Quoc Tran-Dinh, Olivier Fercoq, and Volkan Cevher.
A smooth primal-dual optimization framework for nonsmooth composite convex minimization.
SIAM Journal on Optimization, 28(1):96–134, 2018.

References V

[21] Bang Cong Vu.
A splitting algorithm for dual monotone inclusions involving cocoercive operators.
Advances in Computational Mathematics, 38(3):667–681, 2013.

[22] Zi Xu, Huiling Zhang, Yang Xu, and Guanghui Lan.
A unified single-loop alternating gradient projection algorithm for nonconvex-concave and convex-nonconcave minimax problems.
arXiv preprint arXiv:2006.02032, 2020.

[23] Ming Yan.
A new primal–dual algorithm for minimizing the sum of three functions with a linear operator.
Journal of Scientific Computing, 76(3):1698–1717, 2018.

[24] Alp Yurtsever, Bang Công Vu, and Volkan Cevher.
Stochastic three-composite convex minimization.
In *Advances in Neural Information Processing Systems*, pages 4329–4337, 2016.

[25] Jingzhao Zhang, Hongzhou Lin, Suvrit Sra, and Ali Jadbabaie.
On complexity of finding stationary points of nonsmooth nonconvex functions.
arXiv preprint arXiv:2002.04130, 2020.