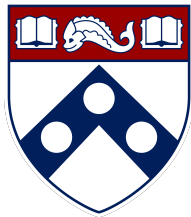# **Dual and primal-dual methods**

Yuxin Chen

Wharton Statistics & Data Science, Fall 2023

Outline

- Dual proximal gradient method

- Primal-dual proximal gradient method

Dual proximal gradient method

Constrained convex optimization

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x})$$
$$\text{subject to} \quad \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} \in \mathcal{C}$$

where f is convex, and $\mathcal{C}$ is convex set

- projection onto such a feasible set could sometimes be highly nontrivial (even when projection onto $\mathcal{C}$ is easy)

Constrained convex optimization

More generally, consider

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) + h(\boldsymbol{A}\boldsymbol{x})$$

where f and h are convex

- computing the proximal operator w.r.t. $\widetilde{h}(\boldsymbol{x}) := h(\boldsymbol{A}\boldsymbol{x})$ could be difficult (even when prox_h is inexpensive)

A possible route: dual formulation

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) + h(\boldsymbol{A}\boldsymbol{x})$$

$$\Updownarrow \text{ add auxiliary variable } \boldsymbol{z}$$

$$\text{minimize}_{\boldsymbol{x}, \boldsymbol{z}} \quad f(\boldsymbol{x}) + h(\boldsymbol{z})$$
$$\text{subject to} \quad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{z}$$

dual formulation:

$$\text{maximize}_{\boldsymbol{\lambda}} \quad \min_{\boldsymbol{x}, \boldsymbol{z}} \underbrace{f(\boldsymbol{x}) + h(\boldsymbol{z}) + \langle \boldsymbol{\lambda}, \boldsymbol{A}\boldsymbol{x} - \boldsymbol{z} \rangle}_{=: \mathcal{L}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\lambda}) \text{ (Lagrangian)}}$$

A possible route: dual formulation

$$\text{maximize}_{\boldsymbol{\lambda}} \quad \min_{\boldsymbol{x},\boldsymbol{z}} \; f(\boldsymbol{x}) + h(\boldsymbol{z}) + \langle \boldsymbol{\lambda}, \boldsymbol{A}\boldsymbol{x} - \boldsymbol{z} \rangle$$

$$\Updownarrow \text{ decouple } \boldsymbol{x} \text{ and } \boldsymbol{z}$$

$$\text{maximize}_{\boldsymbol{\lambda}} \quad \min_{\boldsymbol{x}} \left\{ \langle \boldsymbol{A}^\top \boldsymbol{\lambda}, \boldsymbol{x} \rangle + f(\boldsymbol{x}) \right\} + \min_{\boldsymbol{z}} \left\{ h(\boldsymbol{z}) - \langle \boldsymbol{\lambda}, \boldsymbol{z} \rangle \right\}$$

$$\Updownarrow$$

$$\text{maximize}_{\boldsymbol{\lambda}} \quad - f^*(-\boldsymbol{A}^\top \boldsymbol{\lambda}) - h^*(\boldsymbol{\lambda})$$

where f^* (resp. h^*) is the Fenchel conjugate of f (resp. h)

Primal vs. dual problems

$$\text{(primal)} \quad \text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) + h(\boldsymbol{A}\boldsymbol{x})$$
$$\text{(dual)} \quad \text{minimize}_{\boldsymbol{\lambda}} \quad f^*(-\boldsymbol{A}^\top \boldsymbol{\lambda}) + h^*(\boldsymbol{\lambda})$$

Dual formulation is useful if

- the proximal operator w.r.t. h is cheap (then we can use the Moreau decomposition $\text{prox}_{h^*}(\boldsymbol{x}) = \boldsymbol{x} - \text{prox}_h(\boldsymbol{x})$)

- f^* is smooth (or if f is strongly convex)

Dual proximal gradient methods

Apply proximal gradient methods to the dual problem:

Algorithm 9.1 Dual proximal gradient algorithm

1: **for** $t = 0, 1, \cdots$ **do**
2: $\quad \boldsymbol{\lambda}^{t+1} = \mathsf{prox}_{\eta_t h^*}\left(\boldsymbol{\lambda}^t + \eta_t \boldsymbol{A} \nabla f^*\left(- \boldsymbol{A}^\top \boldsymbol{\lambda}^t \right) \right)$

- let $Q(\boldsymbol{\lambda}) := -f^*(-\boldsymbol{A}^\top \boldsymbol{\lambda}) - h^*(\boldsymbol{\lambda})$ and $Q^{\mathsf{opt}} = \max_{\boldsymbol{\lambda}} Q(\boldsymbol{\lambda})$, then

$$Q^{\mathsf{opt}} - Q(\boldsymbol{\lambda}^t) \lesssim \frac{1}{t} \qquad (9.1)$$

Primal representation of dual proximal gradient methods

Algorithm 9.1 admits a more explicit primal representation

Algorithm 9.2 Dual proximal gradient algorithm (primal representation)

1: **for** $t = 0, 1, \cdots$ **do**
2: $\quad \boldsymbol{x}^t = \arg\min_{\boldsymbol{x}} \ \{f(\boldsymbol{x}) + \langle \boldsymbol{A}^\top \boldsymbol{\lambda}^t, \boldsymbol{x} \rangle\}$
3: $\quad \boldsymbol{\lambda}^{t+1} = \boldsymbol{\lambda}^t + \eta_t \boldsymbol{A} \boldsymbol{x}^t - \eta_t \text{prox}_{\eta_t^{-1} h}(\eta_t^{-1} \boldsymbol{\lambda}^t + \boldsymbol{A} \boldsymbol{x}^t)$

- $\{\boldsymbol{x}^t\}$ is a primal sequence, which is nonetheless *not always feasible*

Justification of the primal representation

By definition of $\boldsymbol{x}^t$,

$$-\boldsymbol{A}^\top \boldsymbol{\lambda}^t \in \partial f(\boldsymbol{x}^t)$$

This together with the conjugate subgradient theorem and the smoothness of f^* yields

$$\boldsymbol{x}^t = \nabla f^*(-\boldsymbol{A}^\top \boldsymbol{\lambda}^t)$$

Therefore, the dual proximal gradient update rule can be rewritten as

$$\boldsymbol{\lambda}^{t+1} = \text{prox}_{\eta_t h^*}(\boldsymbol{\lambda}^t + \eta_t \boldsymbol{A} \boldsymbol{x}^t) \tag{9.2}$$

Justification of primal representation (cont.)

Moreover, from the extended Moreau decomposition, we know

$$\mathsf{prox}_{\eta_t h^*}(\boldsymbol{\lambda}^t + \eta_t \boldsymbol{A} \boldsymbol{x}^t) = \boldsymbol{\lambda}^t + \eta_t \boldsymbol{A} \boldsymbol{x}^t - \eta_t \mathsf{prox}_{\eta_t^{-1} h}(\eta_t^{-1} \boldsymbol{\lambda}^t + \boldsymbol{A} \boldsymbol{x}^t)$$

$$\implies \qquad \boldsymbol{\lambda}^{t+1} = \boldsymbol{\lambda}^t + \eta_t \boldsymbol{A} \boldsymbol{x}^t - \eta_t \mathsf{prox}_{\eta_t^{-1} h}(\eta_t^{-1} \boldsymbol{\lambda}^t + \boldsymbol{A} \boldsymbol{x}^t)$$

Accuracy of the primal sequence

One can control the primal accuracy via the dual accuracy:

Lemma 9.1

Let $\boldsymbol{x_\lambda} := \arg\min_{\boldsymbol{x}} \{f(\boldsymbol{x}) + \langle \boldsymbol{A}^\top \boldsymbol{\lambda}, \boldsymbol{x} \rangle\}$. Suppose f is μ-strongly convex. Then
$$\|\boldsymbol{x}^* - \boldsymbol{x_\lambda}\|_2^2 \leq \frac{2(Q^{\mathsf{opt}} - Q(\boldsymbol{\lambda}))}{\mu}$$

- **consequence:** $\|\boldsymbol{x}^* - \boldsymbol{x}^t\|_2^2 \lesssim 1/t$ (using (9.1))

Proof of Lemma 9.1

Recall that Lagrangian is given by

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\lambda}) := \underbrace{f(\boldsymbol{x}) + \langle \boldsymbol{A}^\top \boldsymbol{\lambda}, \boldsymbol{x} \rangle}_{=: \, \widetilde{f}(\boldsymbol{x}, \boldsymbol{\lambda})} + \underbrace{h(\boldsymbol{z}) - \langle \boldsymbol{\lambda}, \boldsymbol{z} \rangle}_{=: \, \widetilde{h}(\boldsymbol{z}, \boldsymbol{\lambda})}$$

For any $\boldsymbol{\lambda}$, define $\boldsymbol{x}_{\boldsymbol{\lambda}} := \arg\min_{\boldsymbol{x}} \widetilde{f}(\boldsymbol{x}, \boldsymbol{\lambda})$ and $\boldsymbol{z}_{\boldsymbol{\lambda}} := \arg\min_{\boldsymbol{z}} \widetilde{h}(\boldsymbol{z}, \boldsymbol{\lambda})$ (non-rigorous). Then by strong convexity,

$$\mathcal{L}(\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{\lambda}) - \mathcal{L}(\boldsymbol{x}_{\boldsymbol{\lambda}}, \boldsymbol{z}_{\boldsymbol{\lambda}}, \boldsymbol{\lambda}) \geq \widetilde{f}(\boldsymbol{x}^*, \boldsymbol{\lambda}) - \widetilde{f}(\boldsymbol{x}_{\boldsymbol{\lambda}}, \boldsymbol{\lambda}) \geq \frac{1}{2}\mu \|\boldsymbol{x}^* - \boldsymbol{x}_{\boldsymbol{\lambda}}\|_2^2$$

In addition, since $\boldsymbol{A}\boldsymbol{x}^* = \boldsymbol{z}^*$, one has

$$\mathcal{L}(\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{\lambda}) = f(\boldsymbol{x}^*) + h(\boldsymbol{z}^*) + \langle \boldsymbol{\lambda}, \boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{z}^* \rangle = f(\boldsymbol{x}^*) + h(\boldsymbol{A}\boldsymbol{x}^*)$$

$$= F^{\mathsf{opt}} \stackrel{\mathsf{duality}}{=} Q^{\mathsf{opt}}$$

This combined with $\mathcal{L}(\boldsymbol{x}_{\boldsymbol{\lambda}}, \boldsymbol{z}_{\boldsymbol{\lambda}}, \boldsymbol{\lambda}) = Q(\boldsymbol{\lambda})$ gives

$$Q^{\mathsf{opt}} - Q(\boldsymbol{\lambda}) \geq \frac{1}{2}\mu \|\boldsymbol{x}^* - \boldsymbol{x}_{\boldsymbol{\lambda}}\|_2^2$$

as claimed

Accelerated dual proximal gradient methods

One can apply FISTA to dual problem to improve convergence:

Algorithm 9.3 Accelerated dual proximal gradient algorithm

1: **for** $t = 0, 1, \cdots$ **do**

2: $\quad \boldsymbol{\lambda}^{t+1} = \mathsf{prox}_{\eta_t h^*}\Big(\boldsymbol{w}^t + \eta_t \boldsymbol{A} \nabla f^*\big(-\boldsymbol{A}^\top \boldsymbol{w}^t\big)\Big)$

3: $\quad \theta_{t+1} = \frac{1+\sqrt{1+4\theta_t^2}}{2}$

4: $\quad \boldsymbol{w}^{t+1} = \boldsymbol{\lambda}^{t+1} + \frac{\theta_t - 1}{\theta_{t+1}}(\boldsymbol{\lambda}^{t+1} - \boldsymbol{\lambda}^t)$

- apply FISTA theory and Lemma 9.1 to get

$$Q^{\mathsf{opt}} - Q(\boldsymbol{\lambda}^t) \lesssim \frac{1}{t^2} \quad \text{and} \quad \|\boldsymbol{x}^* - \boldsymbol{x}^t\|_2^2 \lesssim \frac{1}{t^2}$$

Primal representation of accelerated dual proximal gradient methods

Algorithm 9.3 admits more explicit primal representation

Algorithm 9.4 Accelerated dual proximal gradient algorithm (primal representation)

1: **for** $t = 0, 1, \cdots$ **do**
2: $\quad \boldsymbol{x}^t = \arg\min_{\boldsymbol{x}} \ f(\boldsymbol{x}) + \langle \boldsymbol{A}^\top \boldsymbol{w}^t, \boldsymbol{x} \rangle$
3: $\quad \boldsymbol{\lambda}^{t+1} = \boldsymbol{w}^t + \eta_t \boldsymbol{A} \boldsymbol{x}^t - \eta_t \mathsf{prox}_{\eta_t^{-1} h} (\eta_t^{-1} \boldsymbol{w}^t + \boldsymbol{A} \boldsymbol{x}^t)$
4: $\quad \theta_{t+1} = \frac{1 + \sqrt{1 + 4\theta_t^2}}{2}$
5: $\quad \boldsymbol{w}^{t+1} = \boldsymbol{\lambda}^{t+1} + \frac{\theta_t - 1}{\theta_{t+1}} (\boldsymbol{\lambda}^{t+1} - \boldsymbol{\lambda}^t)$

Primal-dual proximal gradient method

Nonsmooth optimization

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) + h(\boldsymbol{A}\boldsymbol{x})$$

where f and h are closed and convex

- both f and h might be non-smooth
- both f and h might have inexpensive proximal operators

Primal-dual approaches?

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) + h(\boldsymbol{A}\boldsymbol{x})$$

So far we have discussed proximal methods (resp. dual proximal methods), which essentially updates only primal (resp. dual) variables

Question: can we update both primal and dual variables simultaneously and take advantage of both prox_f and prox_h?

A saddle-point formulation

To this end, we first derive a saddle-point formulation that includes both primal and dual variables

$$\text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) + h(\boldsymbol{Ax})$$

$$\Updownarrow \quad \text{add an auxiliary variable } \boldsymbol{z}$$

$$\text{minimize}_{\boldsymbol{x}, \boldsymbol{z}} \quad f(\boldsymbol{x}) + h(\boldsymbol{z}) \quad \text{subject to } \boldsymbol{Ax} = \boldsymbol{z}$$

$$\Updownarrow$$

$$\text{maximize}_{\boldsymbol{\lambda}} \ \min_{\boldsymbol{x}, \boldsymbol{z}} \ f(\boldsymbol{x}) + h(\boldsymbol{z}) + \langle \boldsymbol{\lambda}, \boldsymbol{Ax} - \boldsymbol{z} \rangle$$

$$\Updownarrow$$

$$\text{maximize}_{\boldsymbol{\lambda}} \ \min_{\boldsymbol{x}} \ f(\boldsymbol{x}) + \langle \boldsymbol{\lambda}, \boldsymbol{Ax} \rangle - h^*(\boldsymbol{\lambda})$$

$$\Updownarrow$$

$$\text{minimize}_{\boldsymbol{x}} \ \max_{\boldsymbol{\lambda}} \ f(\boldsymbol{x}) + \langle \boldsymbol{\lambda}, \boldsymbol{Ax} \rangle - h^*(\boldsymbol{\lambda}) \quad \text{(saddle-point problem)}$$

A saddle-point formulation

$$\text{minimize}_{\boldsymbol{x}} \max_{\boldsymbol{\lambda}} \; f(\boldsymbol{x}) + \langle \boldsymbol{\lambda}, \boldsymbol{A}\boldsymbol{x} \rangle - h^*(\boldsymbol{\lambda}) \qquad (9.3)$$

- one can then consider updating the primal variable $\boldsymbol{x}$ and the dual variable $\boldsymbol{\lambda}$ simultaneously

- we'll first examine the optimality condition for (9.3), which in turn gives ideas about how to jointly update primal and dual variables

Optimality condition

$$\text{minimize}_{\boldsymbol{x}} \ \max_{\boldsymbol{\lambda}} \ f(\boldsymbol{x}) + \langle \boldsymbol{\lambda}, \boldsymbol{A}\boldsymbol{x} \rangle - h^*(\boldsymbol{\lambda})$$

optimality condition:

$$\begin{cases} \mathbf{0} \in & \partial f(\boldsymbol{x}) + \boldsymbol{A}^\top \boldsymbol{\lambda} \\ \mathbf{0} \in & -\boldsymbol{A}\boldsymbol{x} + \partial h^*(\boldsymbol{\lambda}) \end{cases}$$

$$\iff \quad \mathbf{0} \in \left[\begin{array}{cc} & \boldsymbol{A}^\top \\ -\boldsymbol{A} & \end{array} \right] \left[\begin{array}{c} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{array} \right] + \left[\begin{array}{c} \partial f(\boldsymbol{x}) \\ \partial h^*(\boldsymbol{\lambda}) \end{array} \right] =: \mathcal{F}(\boldsymbol{x}, \boldsymbol{\lambda}) \quad (9.4)$$

key idea: iteratively update $(\boldsymbol{x}, \boldsymbol{\lambda})$ to reach a point obeying $\mathbf{0} \in \mathcal{F}(\boldsymbol{x}, \boldsymbol{\lambda})$

How to solve $0 \in \mathcal{F}(x)$ in general?

In general, finding solution to

$$\underbrace{\mathbf{0} \in \mathcal{F}(\boldsymbol{x})}_{\text{called "monotone inclusion problem" if } \mathcal{F} \text{ is maximal monotone}}$$

$$\iff \qquad \boldsymbol{x} \in (\mathcal{I} + \mathcal{F})(\boldsymbol{x})$$

is equivalent to finding fixed points of $\underbrace{(\mathcal{I} + \eta\mathcal{F})^{-1}}_{\text{resolvent of } \mathcal{F}}$, i.e. solutions to
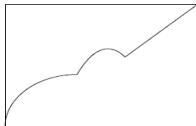
$$\boldsymbol{x} = (\mathcal{I} + \eta\mathcal{F})^{-1}(\boldsymbol{x})$$

This suggests a natural fixed-point iteration / resolvent iteration:

$$\boldsymbol{x}^{t+1} = (\mathcal{I} + \eta\mathcal{F})^{-1}(\boldsymbol{x}^t), \qquad t = 0, 1, \cdots$$

Aside: monotone operators
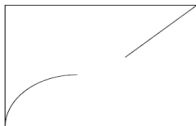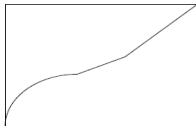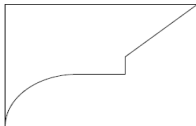
(A) Not monotone.

(B) Monotone but not maximal.

(C) Maximal monotone function.

(D) Maximal monotone but not a function.

- a relation $\mathcal{F}$ is called *monotone* if

$$\langle u - v, x - y \rangle \geq 0, \quad \forall (x, u), (y, v) \in \mathcal{F}$$

- relation $\mathcal{F}$ is called *maximal monotone* if there is no monotone operator that contains it

Proximal point method

$$\boldsymbol{x}^{t+1} = (\mathcal{I} + \eta_t \mathcal{F})^{-1}(\boldsymbol{x}^t), \qquad t = 0, 1, \cdots$$

If $\mathcal{F} = \partial f$ for some convex function f, then this proximal point method becomes

$$\boldsymbol{x}^{t+1} = \mathsf{prox}_{\eta_t f}(\boldsymbol{x}^t), \qquad t = 0, 1, \cdots$$

- useful when $\mathsf{prox}_{\eta_t f}$ is cheap

Recall that we want to solve

$$
\mathbf{0} \in \left[\begin{array}{cc} & \boldsymbol{A}^\top \\ -\boldsymbol{A} & \end{array}\right] \left[\begin{array}{c} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{array}\right] + \left[\begin{array}{c} \partial f(\boldsymbol{x}) \\ \partial h^*(\boldsymbol{\lambda}) \end{array}\right] =: \mathcal{F}(\boldsymbol{x}, \boldsymbol{\lambda})
$$

the issue of proximal point methods: computing $(\mathcal{I} + \eta\mathcal{F})^{-1}$ is in general difficult

Back to primal-dual approaches

observation: practically we may often consider splitting $\mathcal{F}$ into two operators

$$\mathbf{0} \in \mathcal{A}(\boldsymbol{x}, \boldsymbol{\lambda}) + \mathcal{B}(\boldsymbol{x}, \boldsymbol{\lambda})$$

with $\mathcal{A}(\boldsymbol{x}, \boldsymbol{\lambda}) = \begin{bmatrix} & \boldsymbol{A} \\ -\boldsymbol{A}^{\top} & \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{bmatrix}$, $\mathcal{B}(\boldsymbol{x}, \boldsymbol{\lambda}) = \begin{bmatrix} \partial f(\boldsymbol{x}) \\ \partial h^*(\boldsymbol{\lambda}) \end{bmatrix}$
$$(9.5)$$

- $(\mathcal{I} + \eta\mathcal{A})^{-1}$ can be computed by solving linear systems
- $(\mathcal{I} + \eta\mathcal{B})^{-1}$ is easy if prox_f and prox_{h^*} are both inexpensive

solution: design update rules based on $(\mathcal{I} + \eta\mathcal{A})^{-1}$ and $(\mathcal{I} + \eta\mathcal{B})^{-1}$ instead of $(\mathcal{I} + \eta\mathcal{F})^{-1}$

Operator splitting via Cayley operators

We now introduce a principled approach based on operator splitting

$$\text{find} \quad \boldsymbol{x} \quad \text{s.t. } \boldsymbol{0} \in \mathcal{F}(\boldsymbol{x}) = \underbrace{\mathcal{A}(\boldsymbol{x}) + \mathcal{B}(\boldsymbol{x})}_{\text{operator splitting}}$$

let $\mathcal{R}_{\mathcal{A}} := (\mathcal{I} + \eta\mathcal{A})^{-1}$ and $\mathcal{R}_{\mathcal{B}} := (\mathcal{I} + \eta\mathcal{B})^{-1}$ be the resolvents, and $\mathcal{C}_{\mathcal{A}} := 2\mathcal{R}_{\mathcal{A}} - \mathcal{I}$ and $\mathcal{C}_{\mathcal{B}} := 2\mathcal{R}_{\mathcal{B}} - \mathcal{I}$ be the Cayley operators

Lemma 9.2

$$\underbrace{\boldsymbol{0} \in \mathcal{A}(\boldsymbol{x}) + \mathcal{B}(\boldsymbol{x})}_{\boldsymbol{x} \,\in\, \mathcal{R}_{\mathcal{A}+\mathcal{B}}(\boldsymbol{x})} \quad \Longleftrightarrow \quad \underbrace{\mathcal{C}_{\mathcal{A}}\mathcal{C}_{\mathcal{B}}(\boldsymbol{z}) = \boldsymbol{z} \text{ with } \boldsymbol{x} = \mathcal{R}_{\mathcal{B}}(\boldsymbol{z})}_{\textit{it comes down to finding fixed points of } \mathcal{C}_{\mathcal{A}}\mathcal{C}_{\mathcal{B}}} \quad (9.6)$$

$$\boldsymbol{x} \in \mathcal{R}_{\mathcal{A}+\mathcal{B}}(\boldsymbol{x}) \quad \Longleftrightarrow \quad \mathcal{C}_{\mathcal{A}}\mathcal{C}_{\mathcal{B}}(\boldsymbol{z}) = \boldsymbol{z}$$

- **advantage:** allows us to apply $\mathcal{C}_{\mathcal{A}}$ (resp. $\mathcal{R}_{\mathcal{A}}$) and $\mathcal{C}_{\mathcal{B}}$ (resp. $\mathcal{R}_{\mathcal{B}}$) sequentially (instead of computing $\mathcal{R}_{\mathcal{A}+\mathcal{B}}$ directly)

Proof of Lemma 9.2

$$\mathcal{C}_{\mathcal{A}}\mathcal{C}_{\mathcal{B}}(\boldsymbol{z}) = \boldsymbol{z}$$

$$
\begin{align}
& & \boldsymbol{x} &= \mathcal{R}_{\mathcal{B}}(\boldsymbol{z}) & \text{(9.7a)} \\
\Longleftrightarrow & & \widetilde{\boldsymbol{z}} &= 2\boldsymbol{x} - \boldsymbol{z} & \text{(9.7b)} \\
& & \widetilde{\boldsymbol{x}} &= \mathcal{R}_{\mathcal{A}}(\widetilde{\boldsymbol{z}}) & \text{(9.7c)} \\
& & \boldsymbol{z} &= 2\widetilde{\boldsymbol{x}} - \widetilde{\boldsymbol{z}} & \text{(9.7d)}
\end{align}
$$

From (9.7b) and (9.7d), we see that

$$\widetilde{\boldsymbol{x}} = \boldsymbol{x}$$

which together with (9.7d) gives

$$2\boldsymbol{x} = \boldsymbol{z} + \widetilde{\boldsymbol{z}} \tag{9.8}$$

Recall that

$$\boldsymbol{z} \in \boldsymbol{x} + \eta \mathcal{B}(\boldsymbol{x}) \qquad \text{and} \qquad \widetilde{\boldsymbol{z}} \in \boldsymbol{x} + \eta \mathcal{A}(\boldsymbol{x})$$

Adding these two facts and using (9.8), we get

$$2\boldsymbol{x} = \boldsymbol{z} + \widetilde{\boldsymbol{z}} \in 2\boldsymbol{x} + \eta \mathcal{B}(\boldsymbol{x}) + \eta \mathcal{A}(\boldsymbol{x})$$

$$\Longleftrightarrow \qquad \boldsymbol{0} \in \mathcal{A}(\boldsymbol{x}) + \mathcal{B}(\boldsymbol{x})$$

Douglas-Rachford splitting

How to find points obeying $x = \mathcal{C}_\mathcal{A}\mathcal{C}_\mathcal{B}(x)$?

- First attempt: fixed-point iteration

$$z^{t+1} = \mathcal{C}_\mathcal{A}\mathcal{C}_\mathcal{B}(z^t)$$

unfortunately, it may not converge in general

- **Douglas-Rachford splitting**: damped fixed-point iteration

$$z^{t+1} = \frac{1}{2}(\mathcal{I} + \mathcal{C}_\mathcal{A}\mathcal{C}_\mathcal{B})(z^t)$$

converges when a solution to $0 \in \mathcal{A}(x) + \mathcal{B}(x)$ exists!

More explicit expression for D-R splitting

Douglas-Rachford splitting update rule $z^{t+1} = \frac{1}{2}(\mathcal{I} + \mathcal{C}_\mathcal{A}\mathcal{C}_\mathcal{B})(z^t)$ is essentially:

$$
\begin{aligned}
x^{t+\frac{1}{2}} &= \mathcal{R}_\mathcal{B}(z^t) \\
z^{t+\frac{1}{2}} &= 2x^{t+\frac{1}{2}} - z^t \\
x^{t+1} &= \mathcal{R}_\mathcal{A}(z^{t+\frac{1}{2}}) \\
z^{t+1} &= \frac{1}{2}(z^t + 2x^{t+1} - z^{t+\frac{1}{2}}) \\
&= z^t + x^{t+1} - x^{t+\frac{1}{2}}
\end{aligned}
$$

where $x^{t+\frac{1}{2}}$ and $z^{t+\frac{1}{2}}$ are auxiliary variables

or equivalently,

$$
\begin{aligned}
\boldsymbol{x}^{t+\frac{1}{2}} &= \mathcal{R}_{\mathcal{B}}(\boldsymbol{z}^t) \\
\boldsymbol{x}^{t+1} &= \mathcal{R}_{\mathcal{A}}\big(2\boldsymbol{x}^{t+\frac{1}{2}} - \boldsymbol{z}^t\big) \\
\boldsymbol{z}^{t+1} &= \boldsymbol{z}^t + \boldsymbol{x}^{t+1} - \boldsymbol{x}^{t+\frac{1}{2}}
\end{aligned}
$$

Douglas-Rachford primal-dual splitting

$$\text{minimize}_{\boldsymbol{x}} \ \text{max}_{\boldsymbol{\lambda}} \ f(\boldsymbol{x}) + \langle \boldsymbol{\lambda}, \boldsymbol{A}\boldsymbol{x} \rangle - h^*(\boldsymbol{\lambda})$$

Applying Douglas-Rachford splitting to (9.5) yields

$$\boldsymbol{x}^{t+\frac{1}{2}} = \text{prox}_{\eta f}(\boldsymbol{p}^t)$$

$$\boldsymbol{\lambda}^{t+\frac{1}{2}} = \text{prox}_{\eta h^*}(\boldsymbol{q}^t)$$

$$\left[\begin{array}{c} \boldsymbol{x}^{t+1} \\ \boldsymbol{\lambda}^{t+1} \end{array} \right] = \left[\begin{array}{cc} \boldsymbol{I} & \eta \boldsymbol{A}^\top \\ -\eta \boldsymbol{A} & \boldsymbol{I} \end{array} \right]^{-1} \left[\begin{array}{c} 2\boldsymbol{x}^{t+\frac{1}{2}} - \boldsymbol{p}^t \\ 2\boldsymbol{\lambda}^{t+\frac{1}{2}} - \boldsymbol{q}^t \end{array} \right]$$

$$\boldsymbol{p}^{t+1} = \boldsymbol{p}^t + \boldsymbol{x}^{t+1} - \boldsymbol{x}^{t+\frac{1}{2}}$$

$$\boldsymbol{q}^{t+1} = \boldsymbol{q}^t + \boldsymbol{\lambda}^{t+1} - \boldsymbol{\lambda}^{t+\frac{1}{2}}$$

Example

$$\text{minimize}_{\boldsymbol{x}} \quad \|\boldsymbol{x}\|_2 + \gamma\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_1$$

$$\Longleftrightarrow \quad \text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) + g(\boldsymbol{A}\boldsymbol{x})$$

with $f(\boldsymbol{x}) := \|\boldsymbol{x}\|_2$ and $g(\boldsymbol{y}) := \gamma\|\boldsymbol{y} - \boldsymbol{b}\|_1$
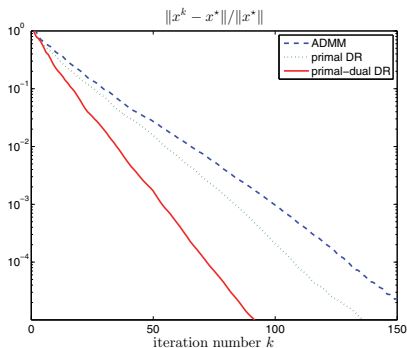
— Connor, Vandenberghe '14

Example

minimize $\quad \|\boldsymbol{Kx} - \boldsymbol{b}\|_1 + \gamma \quad \underbrace{\|\boldsymbol{Dx}\|_{\mathrm{iso}}}_{\text{certain } \ell_2 - \ell_1 \text{ norm}} \qquad$ s.t. $\quad \boldsymbol{0} \leq \boldsymbol{x} \leq \boldsymbol{1}$

$$\Longleftrightarrow \qquad \text{minimize}_{\boldsymbol{x}} \quad f(\boldsymbol{x}) + g(\boldsymbol{Ax})$$

with $f(\boldsymbol{x}) := \mathbb{1}_{\{\boldsymbol{0} \leq \boldsymbol{x} \leq \boldsymbol{1}\}}(\boldsymbol{x})$ and $g(\boldsymbol{y}_1, \boldsymbol{y}_2) := \|\boldsymbol{y}_1 - \boldsymbol{b}\|_1 + \gamma\|\boldsymbol{y}_2\|_{\mathrm{iso}}$
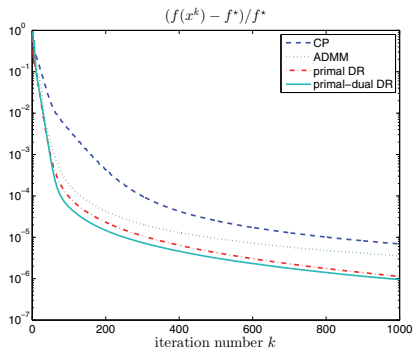
— Connor, Vandenberghe '14

Reference

- "*A First Course in Convex Optimization Theory*," E. Ryu, W. Yin.

- "*Optimization methods for large-scale systems, EE236C lecture notes*," L. Vandenberghe, UCLA.

- "*Convex optimization, EE364B lecture notes*," S. Boyd, Stanford.

- "*Mathematical optimization, MATH301 lecture notes*," E. Candes, Stanford.

- "*First-order methods in optimization*," A. Beck, Vol. 25, SIAM, 2017.

- "*Primal-dual decomposition by operator splitting and applications to image deblurring*," D. O' Connor, L. Vandenberghe, *SIAM Journal on Imaging Sciences*, 2014.

- "*On the numerical solution of heat conduction problems in two and three space variables*," J. Douglas, H. Rachford, *Transactions of the American mathematical Society*, 1956.

Reference

- "*A primer on monotone operator methods*," E. Ryu, S. Boyd, *Appl. Comput. Math.*, 2016.

- "*Large-scale convex optimization: algorithms and analyses via monotone operators*," E. Ryu, W. Yin, *Cambridge University Press*, 2022.